

VŠB – Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra kybernetiky a biomedicínského inženýrství

Pokročilé metody kompenzace šumu pro rozpoznávače řeči  
na bázi virtuální instrumentace

Advanced Anti-noise Speech Recognition Methods Based  
on Virtual Instrumentation

VŠB - Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra kybernetiky a biomedicínského inženýrství

## Zadání diplomové práce

Student:

**Bc. Petr Blaha**

Studijní program:

N2649 Elektrotechnika

Studijní obor:

2612T041 Řídící a informační systémy

Téma:

**Pokročilé metody kompenzace šumu pro rozpoznávače řeči  
na bázi virtuální instrumentace  
Advanced Anti-noise Speech Recognition Methods Based  
on Virtual Instrumentation**

Jazyk vypracování:

čeština

Zásady pro vypracování:

Diplomová práce se věnuje problematice pokročilých metod kompenzace šumu (hluk prostředí) pro rozpoznávače řeči na bázi virtuální instrumentace. Cílem diplomové práce je návrh a realizace virtuálního přístroje pro testování vybraných metod potlačování šumu, určených ke zlepšení výsledné úspěšnosti rozpoznávání řeči. Pro ověření funkčnosti navrženého systému bude nezbytné vytvořit testovací databáze reálných nahrávek včetně návrhu metodiky pro ověření funkčnosti zkoumaných metod.

Postup řešení:

1. Literární rešerše současného stavu problematiky redukce šumu pro rozpoznávače řeči. Klasifikace a popis metod potlačení šumu u řečového signálu (např. lineární a adaptivní filtrace, spektrální odečítání, algoritmy založené na odhadu šumu, ICA, PCA, vlnové transformace, apod.). Rozbor vlivu prostředí a přenosového kanálu na rozpoznávání řeči, problematika detekce řečové aktivity, vlastní rozpoznávání řeči, apod. Rozbor komerčně dostupných rozpoznávačů řeči (Google, Microsoft, Apple, apod.) implementovatelných do vývojového prostředí LabVIEW.
2. Návrh a realizace virtuálního přístroje pro automatizované testování vybraných metod potlačování šumu, určených ke zlepšení výsledné úspěšnosti rozpoznávání řeči (implementace běžně dostupný rozpoznávačů) ve vývojovém prostředí LabVIEW. Součástí navržené aplikace bude systém automatického vyhodnocení úspěšnosti rozpoznávání řeči, umožňující rozsáhlé statické studie (rozpoznávač bez filtrace, s filtrací, různé druhy rušení a mikrofony, různé prostředí, apod.).
3. Tvorba testovací databáze reálných nahrávek, která bude využita pro sérii experimentů pomocí navržené aplikace včetně návrhu metodiky pro ověření funkčnosti zkoumaných metod (např. úspěšnost rozpoznání a skóre rozpoznávače, odstup signálu k šumu, apod.). Databáze by měla být vytvořena použitím různých nahrávacích zařízení (měřicí mikrofony, běžné mikrofony používané v mobilních telefonech, notebookech, tabletech, apod.) v různých prostředích, s různými druhy rušení, různé testované osoby, apod. Předmětem studie bude mimo jiné definování slabých míst v dnešní době využívaných rozpoznávačů řeči.
4. Realizace série experimentálních měření podle navržené metodiky s využitím vytvořeného virtuálního přístroje. Toto měření bude zkoumat vliv testovaných metod pro redukci šumu a jejich parametrů na výslednou úspěšnost rozpoznávání řeči.
5. Diskuze dosažených výsledků.



Seznam doporučené odborné literatury:

- [1] D'SOUZA, Charlton Rodney, et al. Voice Operated Control of a Motor Using LabVIEW. *Electrical and Electronic Engineering*. 2017, vol.7(2),p.60-64. p-ISSN: 2162-9455/e-ISSN: 2162-8459. doi:10.5923/j.eee.20170702.09.
- [2] MOLDŘÍKOVÁ, Zuzana. *Metody potlačení šumu pro rozpoznávací řeči* [online]. Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií, 2014 [cit. 2018-10-19]. Dostupné z: <http://hdl.handle.net/11012/31582>. Diplomová práce. Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií. Ústav biomedicínského inženýrství. Vedoucí práce Jan Odstrčilík.
- [3] SHARMA, Ankush; PERALA, Srinivas; DARSHNI, Priya. Objects Control through Speech Recognition Using LabVIEW. *International Journal of Electronics and Computer Science Engineering*. 2012. V2N1-102-106.ISSN 2277-1956.
- [4] BENESTY, Jacob, et al. *Noise reduction in speech processing*. Springer Science & Business Media, 2009. ISBN 978-3-642-00295-3. doi:10.1007/978-3-642-00296-0.
- [5] DAVIS, Gillian M. *Noise reduction in speech applications*. Boca Raton: CRC Press, 2002. 432p.-163 B/W Illustrations. ISBN 978-0849309496.
- [6] WITTASSEK, Tomáš. *Virtuální instrumentace I: učební text* [CD-ROM]. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, 2014. ISBN 978-80-248-3395-8.
- [7] UHLÍŘ, Jan. *Technologie hlasových komunikací*. Vyd. 1. Praha: Nakladatelství ČVUT, 2007. ISBN 978-80-01-03888-8.

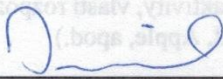
Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **doc. Ing. Radek Martinek, Ph.D.**


Konzultant diplomové práce: Ing. Jan Baroš  
Ing. René Jaroš

Datum zadání: 01.09.2018

Datum odevzdání: 30.04.2019

  
doc. Ing. Jiří Koziorek, Ph.D.  
vedoucí katedry



  
prof. Ing. Pavel Brandštetter, CSc.  
děkan fakulty



### **Prohlášení**

*„Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.“*

V Ostravě dne 30. 4. 2019



.....  
Podpis



## **Poděkování**

Rád bych poděkoval panu doc. Ing. Radku Martinkovi, Ph.D, za odborné rady a informace, které mi poskytnul v průběhu psaní této diplomové práce. Rád bych poděkoval i své rodině a přátelům za podporu během studia i za možnost studovat.

## **Abstrakt**

Tato diplomová práce se věnuje problematice pokročilých metod kompenzace šumu (hluk prostředí) pro rozpoznávače řeči na bázi virtuální instrumentace. Primárním cílem diplomové práce je návrh a realizace virtuálního přístroje pro testování vybraných metod potlačování šumu, určených ke zlepšení výsledné úspěšnosti rozpoznávání řeči. Teoretická část diplomové se věnuje literární rešerši současného stavu problematiky redukce šumu pro rozpoznávače řeči. Experimentální část popisuje návrh a realizaci virtuálního přístroje pro automatizované testování vybraných metod potlačování šumu. V rámci experimentální části byla vytvořena testovací databáze reálných nahrávek, pomocí níž byla provedena série experimentů. Hodnocení kvality filtrace je provedena na základě úspěšnosti rozpoznávače řeči a odstup signálu od šumu.

## **Klíčová slova**

Virtuální Instrumentace, LabVIEW, adaptivní filtrace, metoda nezávislých komponent, detektor řečové aktivity, potlačení šumu, rozpoznávač řeči, odstup signálu od šumu.

## **Abstract**

This thesis deals with advanced noise compensation methods (environment noise) for speech recognition based on virtual instrumentation. The primary goal of this thesis is to design and implement a virtual instrument for testing selected methods of noise suppression designed to improve the resulting speech recognition success. The theoretical part of thesis is devoted to the literature review of the current state of the issue of noise reduction for speech recognizers. The experimental part describes the design and implementation of a virtual instrument for automated testing of selected noise suppression methods. In the experimental part was created a test database of real recordings to perform a series of experiments. Evaluation of quality of filtration is performed on speech recognition success rate and signal to noise ratio.

## **Keywords**

Virtual Instrumentation, LabVIEW, Adaptive Filtration, Independent Component Analysis, Voice Activity Detector, Noise Reduction, Speech Recognition, Signal to Noise Ratio.



## Obsah

Seznam použitých zkratk a symbolů .....	9
Seznam obrázků .....	11
Seznam tabulek .....	13
Úvod.....	14
1    Literární rešerše .....	15
1.1    Vlivy na rozpoznávání řeči .....	16
1.2    Aplikační oblasti rozpoznání řeči.....	19
1.2.1    Uplatnění ve vojenství.....	19
1.2.2    Uplatnění v medicíně .....	19
1.2.3    Uplatnění ve SMART technologiích.....	20
1.2.4    Uplatnění v osobním automobilu .....	20
1.2.5    Využití v každodenním životě a vzdělávání.....	20
1.3    Rozdělení šumů .....	21
1.3.1    Aditivní bílý šum.....	21
1.3.2    Aditivní šum barevný .....	22
1.3.3    Stacionární šum .....	22
1.3.4    Nestacionární šum .....	22
2    Detektor řečové aktivity .....	23
2.1    Energetický detektor .....	23
2.2    Intenzitní detektor .....	23
2.3    Dynamic Time Warping DTW.....	24
2.4    Skryté Markovovy modely HMM.....	26
2.4.1    Rozpoznávání izolovaných slov.....	27
3    Potlačení šumu.....	29
3.1    Adaptivní filtrace.....	30
3.1.1    Adaptivní algoritmus LMS.....	31
3.1.2    Adaptivní algoritmus RLS .....	32
3.2    Slepá separace zdrojů BSS.....	33
3.3    Analýza nezávislých komponent ICA .....	33
3.3.1    Předzpracování ICA .....	35
3.4    Analýza hlavních komponent PCA .....	36
3.4.1    Rozklad pomocí singulárního rozkladu SVD.....	36

4	Experimentální část .....	38
4.1	Měření a použitá zařízení .....	38
4.2	Architektura aplikace .....	44
4.3	Rozhraní zvukové karty .....	45
4.4	Popis výsledné aplikace .....	46
4.4.1	Realizace detektoru řečové aktivity .....	50
4.4.2	Realizace rozpoznávače řeči .....	53
4.5	Testování rozpoznávače WSR .....	57
4.6	Rozbor použitých šumů.....	59
4.7	Výsledky diplomové práce.....	61
4.7.1	Použití v domácnosti .....	63
4.7.2	Kombinace adaptivní filtrace a ICA.....	73
4.7.3	Použití v automobilu .....	77
4.7.4	Kombinace adaptivní filtrace a ICA.....	91
4.7.5	Použití ve výrobní lince.....	95
4.7.6	Kombinace adaptivní filtrace a ICA.....	102
4.7.7	Použití v letadle.....	105
	Závěr .....	108
	Seznam použité literatury.....	110
	Přílohy na CD.....	114



## Seznam použitých zkratk a symbolů

<i>AF</i>	Adaptive Filtation (Adaptivní filtrace)
<i>ASR</i>	Automatic Speech Recognition (Automatické rozpoznání řeči)
<i>BSS</i>	Blind Source Separation (Slepá separace zdrojů)
<i>DTW</i>	Dynamic Time Warping (Dynamické borcení časové osy)
<i>HMM</i>	Hidden Markov Model (Skryté Markovův model)
<i>ICA</i>	Independent Component Analysis (Analýza nezávislých komponent)
<i>LMS</i>	Least Mean Square (Nejmenší průměr kvadrátů)
<i>PCA</i>	Principal Component Analysis (Analýza hlavních komponent)
<i>RLS</i>	Recursive Mean Square (Rekurzivní nejmenší kvadrát)
<i>SNR</i>	Signal to Noise Ratio (Odstup signálu od šumu)
<i>SSNR</i>	Segmental Signal to Noise Ratio (Segmentální odstup signálu od šumu)
<i>VAD</i>	Voice Activity Detector (Detektor řečové aktivity)
<i>WSR</i>	Windows Speech Recognizer (Rozpoznávač řeči od Windows)

<b>A</b>	Ortogonalní matice pro metodu PCA
<b>A<sub>mix</sub></b>	Směsná matice
$\alpha$	Vyhlazovací konstanta
<b>a(n)</b>	Vektor testovaného signálu
$a_{ij}$	Přechodová pravděpodobnost
<b>B</b>	Obraz referenčního slova
<b>b(m)</b>	Vektor referenčního signálu
$b_j(\mathbf{o}_t)$	Výstupní pravděpodobnost
<b>D(A, B)</b>	Vzdálenost mezi obrazy
$E_n$	Krátkodobá energie
$e(n)$	Chybový signál adaptivních filtrů
$g(n, m)$	Funkce lokálního omezení DTW
<b>k(n)</b>	Ziskový vektor
$\lambda$	Faktor zapomínání

$M$	Model
$M_n$	Krátkodobá intenzita
$\mu$	Konvergenční konstanta
$N$	Slovník slov
$n(m)$	Šumový signál
$\mathbf{O}$	Posloupnost vektorů pozorování
$\mathbf{o}_t$	Vektor pozorování
$\mathbf{P}$	Permutační matice
$P(\mathbf{O})$	Pravděpodobnost posloupnosti
$P(\lambda, k)$	Výkonové spektrum šumu
$\Phi_{xx}(n)$	Autokorelační matice
$P(w_i)$	Apriorní pravděpodobnost
$R_{xx}$	Kovarianční matice
$\Psi(n)$	Optimální cesta
$R[0]$	Nultý autokorelační koeficient
$\varrho_d^2$	Odhad hluku
$\mathbf{S}$	Separální matice
$s(t), x(m)$	Čistý řečový signál
$s_i, s_j$	Přechodové stavy
$t$	Časový údaj
$\mathbf{U}$	Ortogonalní matice $m \times m$ pro metodu SVD
$\mathbf{V}$	Ortogonalní matice $n \times n$ pro metodu SVD
$y(m)$	Signál kontaminovaný hlukem
$W$	Vyřčené slovo
$w_i$	Podmíněná pravděpodobnost slov
$w[n]$	Váhovací okno
$\hat{W}$	Rozpoznané slovo
$X$	Stavová sekvence



## Seznam obrázků

Obr. 1: Jednoduchý model mluvčí/rozpoznávač.	16
Obr. 2: Plocha slyšení akustických signálů.	17
Obr. 3: Zachycení odlišnosti podmínek.	18
Obr. 4: Autokorelační funkce a výkonové spektrum bílého šumu.	22
Obr. 5: Detekce slova „Dobrý den“ pomocí VAD.	24
Obr. 6: Funkce DTW pro testovaný a referenční obraz v rovině (n,m).	26
Obr. 7: Markovův model se šesti stavy.	27
Obr. 8: Konfigurace vícekanálového systému.	30
Obr. 9: Blokový diagram adaptivního filtru.	30
Obr. 10: Měřicí mikrofon Rode NT5 a zvuková karta Steinberg UR44.	38
Obr. 11: Směrová charakteristika Rode NT5.	39
Obr. 12: Akustická bezdozvuková komora.	40
Obr. 13: Rozložení mikrofonů v elektrickém vozidle.	40
Obr. 14: Ultralehké letadlo ALTO 912.	41
Obr. 15: Uložení mikrofonu.	41
Obr. 16: Hydraulický lisovací stroj.	42
Obr. 17: Laserové svařovací pracoviště.	42
Obr. 18: Zobrazení rozmístění mikrofonů v bezdozvukové komoře a časové průběhy z jednotlivých pozic při zaznamenávání hluku pračky.	43
Obr. 19: Struktura producent/konzument.	44
Obr. 20: Čelní panel výsledné aplikace.	46
Obr. 21: Zobrazení způsobu normalizace.	48
Obr. 22: Stavový diagram výsledné aplikace.	49
Obr. 23: Vývojový diagram inicializace a průběhu detekce intenzitního detektoru.	51
Obr. 24: Implementace VAD v LabVIEW.	52
Obr. 25: Detekce slova pomocí VAD.	52
Obr. 26: Aplikace rozpoznávače řeči (vlevo) a jeho implementace v Labview.	53
Obr. 27: Engine rozpoznávače řeči WSR.	54
Obr. 28: VB-CABLE Virtual Audio Device.	55
Obr. 29: Implementace virtuálního kabelu.	56
Obr. 30: Implementace Butterworth filtru.	58
Obr. 31: Implementace úpravy amplitudy.	58
Obr. 32: Bílý šum.	59
Obr. 33: Řeč "Turn on the press machine".	59
Obr. 34: Autokorelace a výkonové spektrum pro šum a) myčky, b) pračky, c) ventilátoru.	60
Obr. 35: Algoritmus LMS.	62
Obr. 36: Vliv konvergenční konstanty $\mu$ .	63
Obr. 37: Časové průběhy před a po filtraci za použití LMS algoritmu.	66
Obr. 38: Testované signály v časově-frekvenční oblasti před a po filtraci pomocí LMS algoritmu znázorněné ve spektrogramu.	67
Obr. 39: Časové průběhy před a po filtraci za použití RLS algoritmu.	70

Obr. 40: Testované signály zobrazené v časově-frekvenční oblasti před a po filtraci pomocí RLS algoritmu znázorněné ve spektrogramu.....	71
Obr. 41: Porovnání výsledků filtrace pro algoritmy LMS a RLS. ....	72
Obr. 42: Výpočetní doba adaptivních filtrací.....	73
Obr. 43: Kombinace AF a ICA. ....	73
Obr. 44: Grafické zobrazení úspěšnosti rozpoznávače pro: a) kombinaci LMS a ICA, b) kombinaci RLS a ICA. ....	75
Obr. 45: Porovnání výsledků filtrace pro LMS, RLS a jejich kombinace s metodu nezávislých komponent ICA. ....	76
Obr. 46: Časové průběhy před a po filtraci za použití LMS algoritmu.....	81
Obr. 47: Testované signály zobrazené v časově-frekvenční oblasti před a po filtraci pomocí LMS algoritmu znázorněné ve spektrogramu.....	82
Obr. 48: Časové průběhy před a po filtraci za použití RLS algoritmu.....	87
Obr. 49: Testované signály zobrazené v časově-frekvenční oblasti před a po filtraci pomocí RLS algoritmu znázorněné ve spektrogramu.....	88
Obr. 50: Porovnání výsledků filtrace při zavřených oknech pro: a) SNR, b) SSNR, c) Úspěšnost WSR, pro algoritmy LMS a RLS, použití v automobilu.....	89
Obr. 51: Porovnání výsledků filtrace při otevřeném oknu pro: a) SNR, b) SSNR, c) Úspěšnost WSR, pro algoritmy LMS a RLS, použití v automobilu.....	90
Obr. 52: Grafické zobrazení úspěšnosti rozpoznávače pro: a) kombinaci LMS a ICA, b) kombinaci RLS a ICA (zavřená okna). ....	93
Obr. 53: Grafické zobrazení úspěšnosti rozpoznávače pro: a) kombinaci LMS a ICA, b) kombinaci RLS a ICA (otevřené okno).....	94
Obr. 54: Časové průběhy před a po filtraci za použití LMS algoritmu, v továrně.....	97
Obr. 55: Testované signály zobrazené v časově-frekvenční oblasti před a po filtraci pomocí LMS algoritmu znázorněné ve spektrogramu.....	98
Obr. 56: Časové průběhy před a po filtraci za použití RLS algoritmu, v továrně.....	100
Obr. 57: Testované signály zobrazené v časově-frekvenční oblasti před a po filtraci pomocí RLS algoritmu znázorněné ve spektrogramu.....	101
Obr. 58: Porovnání výsledků filtrace pro: a) SNR, b) SSNR, c) Úspěšnost WSR, pro algoritmy LMS a RLS, použití v továrně.....	102
Obr. 59: Grafické zobrazení úspěšnosti rozpoznávače pro: a) kombinaci LMS a ICA, b) kombinaci RLS a ICA. ....	104
Obr. 60: Časové průběhy před a po filtraci za použití LMS algoritmu, v letadle. ....	106
Obr. 61: Testované signály zobrazené v časově-frekvenční oblasti před a po filtraci pomocí LMS algoritmu znázorněné ve spektrogramu.....	107

## Seznam tabulek

Tab. 1: Paleta dostupných rozhraní (VIs).....	45
Tab. 2: Úspěšnost rozpoznávače řeči, pomocí filtrace.....	57
Tab. 3: Úspěšnost rozpoznávání řeči na příkazu „ <i>Light on</i> “ při změně amplitudy.....	58
Tab. 4: Výsledky filtrace a úspěšnost rozpoznávače pro šum pračky (LMS).....	64
Tab. 5: Výsledky filtrace a úspěšnost rozpoznávače pro šum ventilátoru (LMS).....	64
Tab. 6: Výsledky filtrace a úspěšnost rozpoznávače pro šum myčky (LMS).....	65
Tab. 7: Výsledky filtrace a úspěšnost rozpoznávače pro šum pračky (RLS).....	68
Tab. 8: Výsledky filtrace a úspěšnost rozpoznávače pro šum ventilátoru (RLS).....	68
Tab. 9: Výsledky filtrace a úspěšnost rozpoznávače pro šum myčky (RLS).....	69
Tab. 10: Úspěšnost rozpoznávače řeči pro LMS a LMS + ICA.....	74
Tab. 11: Úspěšnost rozpoznávače řeči pro RLS a RLS + ICA.....	74
Tab. 12: Výsledky filtrace a rozpoznávače pro rychlost 80 km/h, zavřená okna (LMS).....	77
Tab. 13: Výsledky filtrace a rozpoznávače pro rychlost 100 km/h, zavřená okna (LMS).....	78
Tab. 14: Výsledky filtrace a rozpoznávače pro rychlost 130 km/h, zavřená okna (LMS).....	78
Tab. 15: Výsledky filtrace a rozpoznávače pro rychlost 80 km/h, otevřené okno (LMS).....	79
Tab. 16: Výsledky filtrace a rozpoznávače pro rychlost 100 km/h, otevřené okno (LMS).....	79
Tab. 17: Výsledky filtrace a rozpoznávače pro rychlost 130 km/h, otevřené okno (LMS).....	80
Tab. 18: Výsledky filtrace a rozpoznávače pro rychlost 80 km/h, zavřená okna (RLS).....	84
Tab. 19: Výsledky filtrace a rozpoznávače pro rychlost 100 km/h, zavřená okna (RLS).....	84
Tab. 20: Výsledky filtrace a rozpoznávače pro rychlost 130 km/h, zavřená okna (RLS).....	85
Tab. 21: Výsledky filtrace a rozpoznávače pro rychlost 80 km/h, otevřené okno (RLS).....	85
Tab. 22: Výsledky filtrace a rozpoznávače pro rychlost 100 km/h, otevřené okno (RLS).....	86
Tab. 23: Výsledky filtrace a rozpoznávače pro rychlost 130 km/h, otevřené okno (RLS).....	86
Tab. 24: Úspěšnost rozpoznávače řeči pro LMS a LMS + ICA (zavřená okna).....	91
Tab. 25: Úspěšnost rozpoznávače řeči pro RLS a RLS + ICA (zavřená okna).....	91
Tab. 26: Úspěšnost rozpoznávače řeči pro LMS a LMS + ICA (otevřené okno).....	92
Tab. 27: Úspěšnost rozpoznávače řeči pro RLS a RLS + ICA (otevřené okno).....	92
Tab. 28: Výsledky filtrace a úspěšnost rozpoznávače pro hluk svářecího stroje (LMS).....	96
Tab. 29: Výsledky filtrace a úspěšnost rozpoznávače pro hluk lisovacího stroje (LMS).....	96
Tab. 30: Výsledky filtrace a úspěšnost rozpoznávače pro hluk svářecího stroje (RLS).....	99
Tab. 31: Výsledky filtrace a úspěšnost rozpoznávače pro hluk lisovacího stroje (RLS).....	99
Tab. 32: Úspěšnost rozpoznávače řeči pro LMS a LMS + ICA.....	103
Tab. 33: Úspěšnost rozpoznávače řeči pro RLS a RLS + ICA.....	103



# Úvod

V dnešní době se vyskytuje řada převratných technologických změn, která souvisí s vývojem digitálních a informačních technologií určených ke zpracování a užívání informací. Typickým příkladem užívání informace je komunikace v reálném čase v oblasti mobilních technologií, která se stále rychle vyvíjí. V každé reálné aplikaci zpracovávání řečového signálu, musíme počítat s přítomností šumu. Pojem šum bude představovat hluk prostředí při zaznamenávání řeči. Automatické rozpoznávání řeči se v současnosti stále více využívá v reálných systémech. Robustnost, která je nutná u těchto systémů vůči rušení je v této oblasti výzkumu jednou z nejvíce studovaných problematik. Je nutné kvantifikovat úroveň šumového pozadí v řečovém signálu.

První kapitola je zpracována jako rešerše problematiky pro automatické rozpoznávání řeči. Jsou zde popsány konkrétní faktory, které mají vliv na kvalitu řeči. Ve druhé části této kapitoly se nachází popis aplikačních oblastí, ve kterých se rozpoznávání řeči používá. Podle těchto oblastí použití byly stanoveny příklady pro konkrétní oblasti, kterými se tato diplomová práce bude dále zabývat. Poslední část tvoří popis užitých hluků.

Popis detektoru řečové aktivity a jeho princip funkčnosti se nachází ve druhé kapitole. Detektor řeči se používá i u telefonních spojů, kde lze vynechat oblasti řeči, ve kterých se nenachází řeč. Bude to mít za následek, že pro spoluúčastníka hovoru, dojde k nepřírozenému ztišení reproduktoru. Je zde uvedeno, jaké druhy detektorů řeči se běžně užívají. Dále jsou zde popsány detektory na stylu dynamického borcení časové osy a skrytých Markovových modelů.

Třetí kapitola se zabývá popisem metod, které se používají pro potlačování šumu (hluku okolí) v řečovém signálu. Je zde popsána adaptivní filtrace, která slouží adaptivnímu potlačování šumu a rušení. V praxi jsme často postaveni před problém filtrace v neznámém prostředí, ve kterém je obtížná předběžná identifikace a nelze předpovídat vývoj do budoucna. Dále je zde popsána analýza nezávislých komponent, která nepracuje na principu filtrace. Jedná se o metodu, která se zabývá separací signálů, které spolu byly určitým způsobem smíchány. Jako příklad se uvádí tzv. „Coctail – party problém“, který popisuje problém smíchání řeči dvou mluvčích.

Ve čtvrté kapitole diplomové práce se nachází popis aplikací, které byly vytvořeny ve vývojovém prostředí LabVIEW. Jedná se o popis softwarové a hardwarové části zpracovaných aplikací a popis jednotlivých prostředí ve kterém probíhalo měření. V první řadě probíhalo měření v částečně bezdozvukové komoře, aby byly nahrány referenční nahrávky řeči. Poté probíhalo měření v oblastech, ve kterých je možné použít hlasové ovládání. Jednalo se celkem o čtyři různé oblasti, aby bylo získáno dostatečné množství různých druhů hluků. Dále jsou zde popsány výsledky filtrace pomocí adaptivních filtrů a metody nezávislých komponent, z pohledu hodnot odstupu signálu od šumu a úspěšnosti rozpoznávače řeči.

Poslední kapitola se věnuje zhodnocení jednotlivých výsledků diplomové práce. Je zde uveden postup celou diplomovou prací a zhodnocení použitých metod, které byly vybrány pomocí literární rešerše. Následně jsou zde popsány výsledky, kterých bylo dosaženo pomocí vybraných metod. Tyto metody pro kompenzaci šumu, byly použity samostatně, i dále i v jejich vzájemné kombinaci. Výsledky jsou uvedeny pro jednotlivé oblasti, ve kterých měření probíhalo. Jednalo se o použití v domácnosti, v osobním automobilu, v továrně a v letadle.

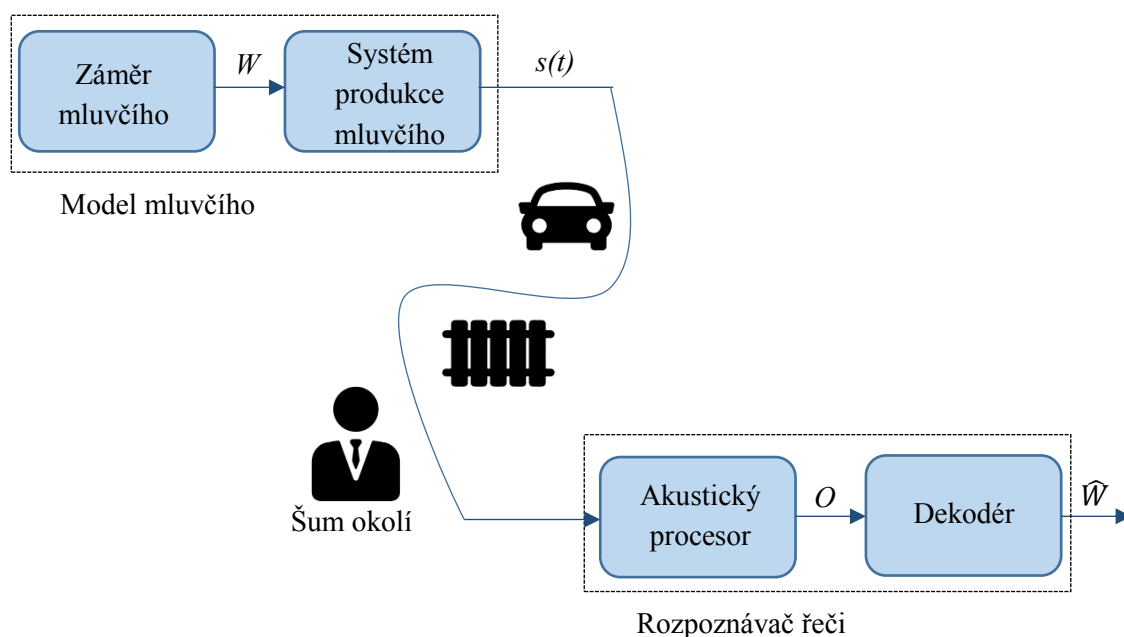
# 1 Literární řešení

Metody rozpoznávání řeči se již dokázaly prosadit v mnoha aplikačních oblastech, jako je např. hlasové ovládání v inteligentních domácnostech, v osobních automobilech, lékařství, vojenství apod. Největším problémem při rozpoznávání řeči je hluk, který se v našem okolí nachází, a proto musíme klást velkou pozornost na jeho potlačení. Tento hluk by velkým způsobem znehodnocoval kvalitu řeči. Proto je potřeba se zabývat filtrací rušení (hluk okolí). Této problematice se pozorně věnuje diplomová práce.

Automatické rozpoznání řeči ASR (Automatic Speech Recognition) představuje postup, kdy jsou jednotlivé řečové signály převáděny na posloupnosti slov. Hlavním cílem automatických rozpoznávačů řeči je přiblížit účinnost rozpoznávání lidské bytosti [1]. Jedná se o soubor algoritmů, které umožňují stroji porozumět lidskému mluvenému projevu. Existuje mnoho aplikací, které využívají právě ASR jako je například: hlasové vyhledávání, diktování krátkých textů, ovládání systémů ve vozidle, osobní asistent, hry nebo hlasové ovládání domácnosti a další [2]. Automatické rozpoznání řeči je považováno za velmi komplikovaný úkol. Vědci musí zvážit zpracování signálu, rozpoznání vzorů, lingvistiku, programování. Většina vědců uplatňuje monolitický přístup k jednotlivým problémům [3]. Mnoho faktorů ovlivňuje složitost návrhu ASR, jako jsou izolovaná nebo propojená slova, spojitá řeč, závislý a nezávislý mluvčí nebo malá slovní zásoba [4].

Zatímco současné systémy ASR pracující v bezhlukém prostředí, dosahují velmi vysokých přesností, v prostředí s hlukem v pozadí mohou selhat [5][6]. Pro zlepšení výkonu ASR v hlučném prostředí se musí posílit extrakce funkcí, transformovat akustické modely nebo vycvičit obecnější modely [7]. Transformace si klade za cíl najít vztah mezi vlastnostmi hluku a vyškolenými modely. Principem modelu je trénovat modely na všechny možné vlastnosti hluku, s tím že mají více řečových variací v tréninkovém procesu [7].

Řeč je základní a hlavní forma komunikace mezi lidmi. Komunikace probíhá mezi různými řečníky a posluchači. Tento proces komunikaci si můžeme představit jako vysílače a přijímače [3]. Tedy vysílače neboli řečníky musí vyslovit zvukové vlny, které v prostředí představují řečový signál. Na tyto vlny se v prostředí komunikace superponuje hluk okolí. Přijímač neboli posluchač musí tuto vyřčenou řeč rozpoznat a plně jí porozumět. Posluchače si lze představit jako rozpoznávače řeči, který musí z řečového signálu odfiltrout řeč od hluku.



Obr. 1: Jednoduchý model mluvího/rozpznávač.

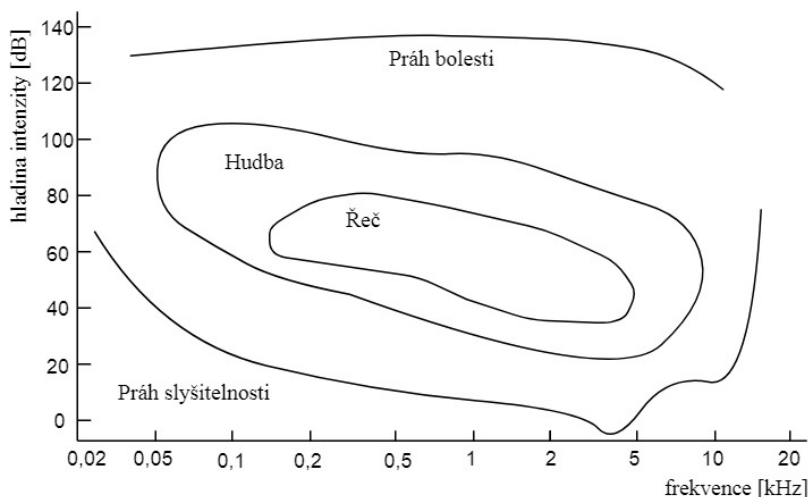
Na Obr. 1 je znázorněn jednoduchý model [1] kdy mluvčí vysloví svoji výpověď  $W$ , která je dále zakódována do akustického signálu  $s(t)$  ke kterému se superponuje hluk (šum) okolí. Rozpoznávač řeči poté musí analyzovat řečový signál  $s(t)$  v časové i spektrální doméně pro získání odlišných pozorovacích znaků  $O$ . V posledním kroku by měl poté dekodér rozpoznat nepravděpodobnější slovní sekvenci  $\hat{W}$  [1].

## 1.1 Vlivy na rozpoznávání řeči

Na kvalitu rozpoznávání řeči má vliv řada faktorů jako je rozmanitost jazyka, která může být rozdělena:

- **Kulturní** – je označováno úrovní vzdělání, a to má silný vliv na styl mluvení.
- **Fyzická** – je označována stavbou těla jedince, a to se odlišuje mluvením a jasností řeči
- **Psychologická** – každá osoba má různé styly mluvení v závislosti na emocionálním stavu, záměru a postoji.

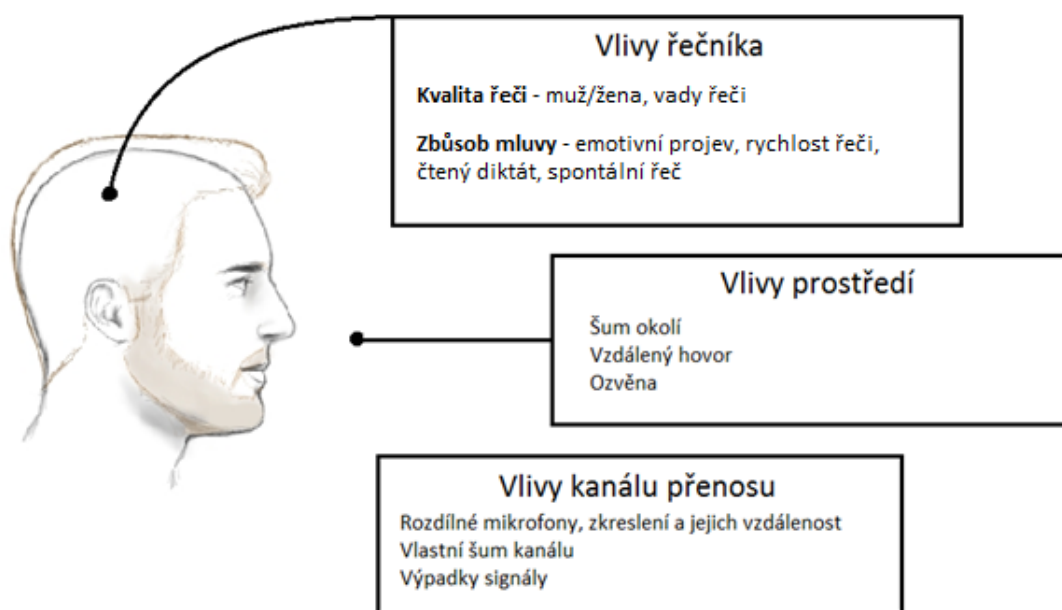
Z frekvenčního hlediska se řeč nachází v rozmezí 180 Hz až 6000 Hz. Nejdůležitější složky, které zajišťují srozumitelnost se nacházejí v oblasti od 300 Hz do 5000 Hz [9]. Pro nejběžnější komunikaci stačí přenosové pásmo v rozmezí 300 Hz až 2500 Hz [10]. Obr. 2 zobrazuje vymezení hranice řeči.



Obr. 2: Plocha slyšení akustických signálů.

Na nižších frekvencích můžeme slyšet spíše samohlásky a na vyšších souhlásky. Úspěšné rozpoznání řeči je zajištěno, jsou-li správně rozeznány vyřčené fráze. To je ale ovlivněno různými faktory, jako je spolehlivost detekčního systému nebo třeba intonace a správná výslovnost řečníka [9]. Rozpoznání řeči lze využít jak v běžném životě, jako jsou osobní automobily nebo mobilní telefony, kde můžeme rozpoznávání řeči využít k ovládání jednotlivých funkcí, tak i například ve zdravotnictví, vojenství či robotice.

Velmi častým jevem, který provází systémy ASR je, že jejich funkce je ovlivněna reálným prostředím. Nemáme tedy často k dispozici pouze samotný řečový signál, ale signál, jehož průběh je kontaminován vnějším prostředím (hlukem). Nejčastěji je to způsobeno tím, že se liší podmínky, ve kterých byl systém trénován a ve kterých bude následně provozován [11]. Systémy automatického rozpoznávání řeči, jejichž přesnost se téměř nemění při různých okolních podmínkách, nazýváme robustními systémy [11]. Této robustnosti nedosáhne systém automaticky, ale je třeba systémy na tyto různá prostředí přizpůsobit. Na následujícím Obr. 3 jsou uvedeny hlavní odlišnosti, ke kterým může docházet v reálném prostředí, ve kterém bude tento systém pracovat.



Obr. 3: Zachycení odlišnosti podmínek.

Předchozí obrázek zobrazuje tři hlavní odlišnosti v řeči, které mají vliv na její rozpoznávání. Prvním z nich je vliv řečníka. Jednou z hlavních disproporcí mezi podmínkami trénování a činnosti systému rozpoznávání řeči je samotný řečník. Pro úlohy, kdy je požadována nezávislost funkce systému na řečníkovi. Tedy když má být nasazován v úlohách, kdy má rozeznávat hlasy jednotlivých řečníků. Musí být v procesu trénování využito velké množství různých řečníků, aby byla co nejlépe zachycena jejich artikulace, dialekty apod. Dále je důležité, aby při trénování byly zastoupeny v testovací množině jak mužské hlasy, tak i ženské. Tedy čím větší množinu hlasů se pro trénování podaří shromáždit, tím bude výsledný systém spolehlivější v provozních podmínkách. V úlohách, kdy řeč probíhá v prostředí, které je zatíženo hlukem, dochází k Lombardovu efektu [12]. Tento efekt se projevuje výraznými změnami jak akustických, tak artikulačních [9], [13].

Druhým vlivem je akustické prostředí. Funkci automatického rozpoznávání výrazně ovlivňuje akustické prostředí, ve kterém je výsledný systém provozován. Jde o hluk prostředí, který je nekorelovaný s řečí. Tyto hluky mohou mít stacionární či nestacionární charakter. Jsou vysvětleny v následující kapitole. Často se objevují šумы, které jsou nekorelované s řečí a které mají také vliv na funkci rozpoznávače. Tyto druhy šumů vyplývají nejčastěji z akustických odrazů od zdí nebo jiných objektů [9].

Posledním vlivem uvedeným na obrázku je vliv přenosového kanálu. Jednou z nejdůležitějších komponent přenosového kanálu je použitý mikrofon. Mikrofony transformují řečové kmity na elektrický signál různými způsoby a mají často velmi odlišné přenosové a směrové charakteristiky. Můžeme zjistit že i při použití stejných mikrofonů dosáhneme proměnlivých přenosových funkcí v závislosti na vzdálenosti mikrofonu od řečníka. Často ani není možné zajistit, aby systém rozpoznávání řeči byl trénován s využitím stejného mikrofonu. Nevíme totiž, který mikrofon bude



poté využíván v reálném prostředí. Významný vliv na výsledný řečový signál má i vlastní přenosový kanál. Především je to frekvenční šířka kanálu, nevyrovnaná přenosová charakteristika a rušení, které výrazně ovlivňuje kvalitu řečového signálu. Jinou kvalitu rozpoznávání dosáhneme při použití mikrofону a přenosového kanálu, které zajišťují téměř dostatečnou šířku pásma (22kHz) nebo při využití telefonního kanálu, kdy bývá šířka pásma kolem 3kHz [9].

## 1.2 Aplikační oblasti rozpoznání řeči

Rozpoznávání řeči umožňuje uživateli komunikovat například s mobilním telefonem jen za pomoci řeči. Jedná se tedy o aplikaci, která vyřčená slova nebo fráze, převede na text nebo vykoná co se právě požaduje. Rozpoznávání řeči je navrženo za účelem vytvoření textu z řeči, bez nutnosti použití klávesnice. Pro rozpoznávání řeči existují dva typy zpracování [14]:

- **Front-end** (závislé na uživateli): řeč je v reálném čase přeměněna na text, tedy není potřeba použití transkriptoru.
- **Back-end** (nezávislé na uživateli): slova jsou zaznamenána v digitální podobě a poté zpracována počítačem, po kterém je text dále zpracován korektním editorem.

### 1.2.1 Uplatnění ve vojenském

Pilot, který letá ve vysoce výkonných letounech, má oči i ruce stále zaneprázdněny ovládáním stroje a automatické rozpoznávání řeči je pro něj velice výhodné. Pilot může nastavovat rádiové frekvence nebo ovládat výběr zbraní, jen za pomoci hlasových příkazů. Díky tomu se dokáže lépe soustředit na letoun v bojových situacích. Zvláštní pozornost je věnována americkému programu rozpoznávání řeči pro letadla. Jedná se o pro pokročilé technologie integrace bojových technologií AFTI [15] a program ve Francii, který instaluje systémy rozpoznávání řeči pro letouny Mirage [16], [17].

### 1.2.2 Uplatnění v medicíně

Rozpoznávání řeči se v medicíně na základě přehledu literatury uplatňuje v různých kategoriích, jako jsou [18]:

- Systémy na diktování.
- Interaktivní systémy.
- Systémy pro řízení zdravotnického vybavení.

Hlavním zaměřením rozpoznávání řeči ve zdravotnických aplikacích jsou systémy, které slouží k záznamu dat do lékařských zpráv (elektronické záznamy pacientů) [20], [21]. V [20] provedli Rodger a Pendharkar studii, které se účastnilo třicet tři lékařů, aby dokončili řadu úloh v oblasti

dokumentace pomocí hlasového ovládání a poté i pomocí klávesnice a myši. Výsledná přesnost byla pro hlasové ovládání od 85 % do 99 % a pro klávesnici 100 %. Bylo zjištěno, že ovládání pomocí řeči trvá přibližně o polovinu kratší dobu než při použití klávesnice a myši a také že pohlaví a seznámení se s aplikací výrazně ovlivňuje přesnost při rozpoznávání řeči.

### **1.2.3 Uplatnění ve SMART technologiích**

Hlavním rysem inteligentního domova je schopnost ovládat funkce domova a jeho spotřebičů, jako aplikaci funkce příkazu a kontroly [19], [22]. Většina aplikací příkazu a kontroly má jen omezenou slovní zásobu, která bývá kolem padesáti slov. Modul detekce řeči je výchozím bodem celého systému. Monitoruje zvuk ze všech mikrofónů a při detekci lidské řeči zaznamenává zvukovou nahrávku [23]. Nahrávka je předána do modulu pro rozpoznávání řeči. Společně s nahrávkou se předává i index mikrofónu, aby byla zjištěna poloha uživatele. Budeme-li mít například příkaz „Ahoj Alex, zapni světlo v koupelně“, je tento příkaz tvořen rámcovým slovem „Ahoj Alex“, dále operací, co má udělat „zapni“, umístěním, kde to má udělat „v koupelně“ a daným zařízením „světlo“ [24], [25].

### **1.2.4 Uplatnění v osobním automobilu**

Pomocí ručního ovládání vstupu, například pomocí ovládání prstů na volantu, povolí systém rozpoznávání řeči, a řidiči je signalizována zvuková výzva. Po zvukové výzvě má systém "poslechové okno", během něhož může přijmout řečový signál pro rozpoznávání. Jednoduché hlasové příkazy mohou být použity například k inicializaci telefonních hovorů, výběru rozhlasových stanic nebo přehrávání hudby z mobilního telefonu. Funkce rozpoznávání hlasu se liší mezi značkou a modelem. Některé z nejnovějších modelů vozidel nabízejí rozpoznávání řeči přirozeného jazyka namísto pevné sady příkazů, což umožňuje řidiči používat plné věty a běžné fráze. U takových systémů tedy není potřeba, aby uživatel zapamatoval soubor pevných příkazových slov [26], [27].

### **1.2.5 Využití v každodenním životě a vzdělávání**

Pro výuku jazyků může být rozpoznávání řeči užitečné. Může naučit správnou výslovnost, kromě toho, že pomáhá člověku rozvíjet plynulost s mluvením.

Studenti, kteří jsou slepí, mohou tuto technologii rozpoznávání řeči využívat k předávání slov a poté slyšet počítač přednášet je. Stejně tak mohou používat počítač jen za pomoci svého hlasu, kdy hlasovými příkazy jej dokážou ovládat, aniž by se museli dívat na obrazovku a klávesnici. Dalším využitím pro studenty, kteří jsou tělesně postižení a potýkají se s problémem, který jim nedovoluje svépomocí pracovat na školních úkolech, je využití technologie převodu řeči na text a tím si usnadnit svoji práci [28], [29].

Svět rozpoznávání řeči se rychle mění a vyvíjí. Včasná aplikace technologie dosáhla různého stupně úspěchu. Příslib pro budoucnost je výrazně vyšší výkon pro téměř každou oblast technologie rozpoznávání řeči, s větší robustností pro reproduktory, zvuky pozadí atd. To nakonec povede ke spolehlivým a robustním hlasovým rozhraním pro každou telekomunikační službu, která je nabízena, a tím je všeobecně dostupný. Hlasové ovládání v jednotlivých oblastech má své výhody i nevýhody.

Výhody rozpoznávání řeči:

- Řeč je přirozený způsob komunikace a není nutné ovládání pomocí klávesnice anebo dálkového ovladače.
- Není potřeba školení.
- Úspora času.
- Řeč zajišťuje flexibilitu.
- Data jsou opakovaně použitelná.

Nevýhody rozpoznávání řeči:

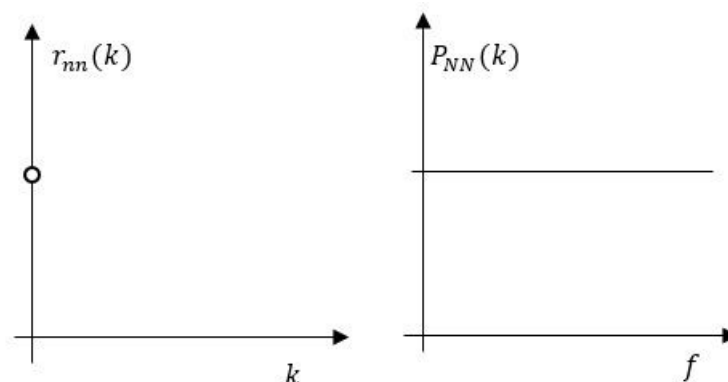
- Pokud se v prostředí vyskytuje hluk, počet chybně rozpoznávaných slov se zvýší.
- Vzdálenost umístění mikrofону od řečníka (s rostoucí vzdáleností roste chybovost).

## 1.3 Rozdělení šumů

Zdroje šumu mohou být rozděleny na aditivní šумы a konvoluční šумы [30]. Aditivní šумы lze dále rozdělit podle spektrálních vlastností na šum bílý a šum barevný. Tyto šумы nejsou korelované se signálem. Konvoluční šумы jsou způsobeny například různými odrazy v prostředí šíření, rušením způsobeným přenosem signálu nebo vlastnostmi mikrofону a tím zaváděny nechtěné artefakty do originálního signálu řeči. Tento šum je korelovaný s originálním signálem, více v [9].

### 1.3.1 Aditivní bílý šum

Tento druh šumu se v reálném prostředí nevyskytuje, ale používá se v modelech reálných systémů, kdy se pomocí tohoto šumu aproximuje širokopásmové rušení. Aditivní bílý šum má konstantní výkonovou spektrální hustotu, která se rovná jeho rozptylu. Dalším důležitým parametrem, který tento šum má je jeho střední hodnota, která se ve většině případů uvažuje nulová. Nulové jsou také koeficienty autokorelační funkce, kromě hodnoty  $R[0]$ , která je rovna rozptylu (Obr. 4). Více v [9], [31].



Obr. 4: Autokorelační funkce a výkonové spektrum bílého šumu.

### 1.3.2 Aditivní šum barevný

Tento nekorelovaný šum má jiné rozložení energie ve spektru než předchozí bílý šum. Oproti bílému šumu se barevný šum liší v tom, že má nenulové i další koeficienty autokorelační funkce než jen  $R[0]$  [9].

Šumový signál se může dále rozdělit na stacionární a nestacionární.

### 1.3.3 Stacionární šum

Šum stacionární má výkonovou spektrální hustotu v čase téměř konstantní a může být způsoben například zvukem počítače, deště nebo jej představuje zvuk nastartovaného auta. Většinou se v reálných aplikacích předpokládá, že každý neznámý šum je stacionárního charakteru.

### 1.3.4 Nestacionární šum

Oproti stacionárnímu šumu je tento nestacionární šum charakterizován tím, že se jeho spektrální charakteristiky s časem mění a nejsou tedy konstantní, jak tomu bylo u stacionárního šumu. Zatímco u stacionárního šumu bylo uvedeno, že jej představuje například zvuk nastartovaného automobilu, nestacionární šum způsobí zvuk projíždějícího automobilu, tedy zvuk, který se postupně s časem mění. Dalšími příklady může být například štěkot psa, psaní na počítačové klávesnici nebo zvuk bouchání kladiva.

V dnešní době se poměrně často můžeme setkat s šumem nestacionárním, jelikož se v průběhu rozhovoru často mění naše akustické pozadí, např. při komunikaci pomocí mobilního telefonu během cesty po rušné ulici. Je třeba podotknout, že šum má nepříznivý vliv na řečový signál. Nepříznivý vliv je způsoben při komunikaci v hlučném prostředí, kdy se snažíme mluvit co nejeфекtivněji. Dochází tím ke změně akustických parametrů řeči. Tomuto jevu, kterému se řečník snaží přizpůsobit při komunikaci v hlučném prostředí, se nazývá Lombardův efekt [12].

## 2 Detektor řečové aktivity

Detektor řečové aktivity VAD (Voice Activity Detector) je systém, do kterého vstupuje čistý řečový signál, nebo směs řeči a hluku. Pomocí VAD získáme ze zvukové nahrávky jen ty segmenty, ve kterých se nachází řeč. Typickými parametry pro detekování může být například energie nebo intenzita.

Detektory VAD obvykle musí splňovat následující požadavky [32]:

- Měl by být realizovaný ve frekvenční oblasti.
- Informaci o přítomnosti řeči by měl získávat pouze v krátkých aktuálních segmentech a pracovat v reálném čase.
- Detektor by měl dosahovat velké pravděpodobnosti správné detekce řeči, pauzy při relativně malém poměru signál/šum.

### 2.1 Energetický detektor

Energetické detektory jsou jedny z nejstarších, ale stále používaných detektorů. Popularita těchto detektorů je dána především jejich nízkou výpočetní náročností. Jejich funkčnost je založena na principu detekce vyšší hladiny energie v segmentech, kde se nachází řeč oproti segmentům bez řečového signálu. Tento detektor zjišťuje energii signálu pro každý segment nahrávky pomocí vzorce [33]:

$$E_n = \sum_{n=-\infty}^{\infty} (y[m]w[n-m])^2, \quad (1.1.)$$

kde  $y(m)$  je signál kontaminovaný hlukem okolí a  $w[n]$  je typ váhovacího okna.

Úseky řeči mají vyšší hladinu energie, a proto energetický detektor má výbornou funkčnost. Nevýhodou těchto detektorů je pokles funkčnosti při intenzivnějším šumu v pozadí.

### 2.2 Intenzitní detektor

Dalším parametrem oproti energii může být intenzita. Výhodou tohoto parametru je, že není příliš citlivý na velké změny úrovně signálu jako energie. Krátkodobá intenzita se dá vypočítat podle vztahu [33]:

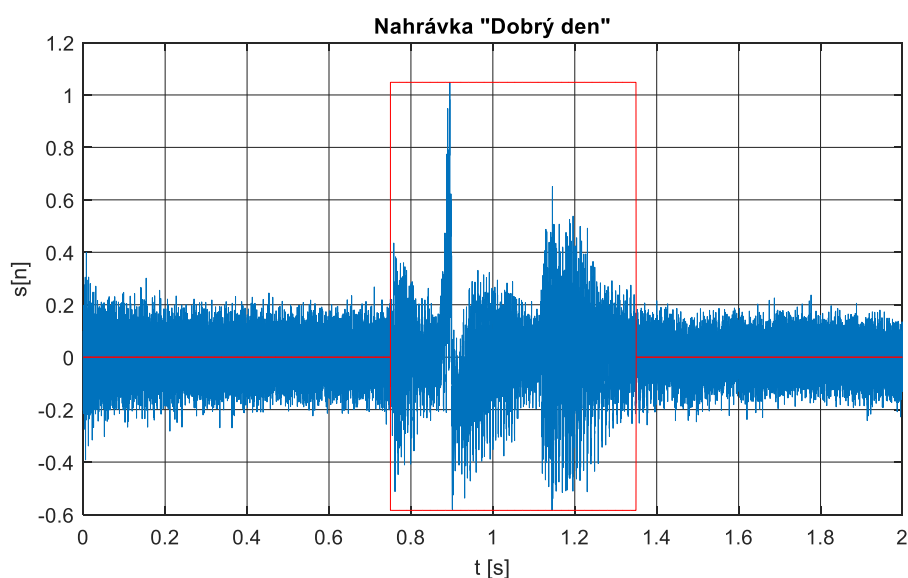
$$M_n = \sum_{n=-\infty}^{\infty} |y[m]w[n-m]|. \quad (1.2.)$$



Funkce detektoru řečové aktivity se dá shrnout do dvou kroků:

- V prvním kroku, inicializační fáze, se provede inicializace signálu. Jedná se o pár segmentů na začátku signálu. V těchto segmentech by neměla být přítomna řeč. Z vybraných segmentů se určí prahová hodnota pro detekci.
- Ve druhém kroku, detekční fáze, detektor řečové aktivity porovnává jednotlivé hodnoty segmentů s prahovou hodnotou a určuje se, ve kterých segmentech se nachází řeč. Jeli hodnota segmentu větší než prahová hodnota. Bude daný segment označen jako ten, ve kterém se nachází řeč, v opačném případě se jedná o řečovou pauzu.

Na Obr. 5 je zobrazena detekce pomocí detektoru řečové aktivity při použití parametru intenzity ze zašuměného signálu.



Obr. 5: Detekce slova „Dobry den“ pomocí VAD.

## 2.3 Dynamic Time Warping DTW

Metoda borcení časové osy (Dynamic Time Warping) DTW, využívá principu dynamického programování. Tato metoda sice patří mezi starší, avšak pro svou jednoduchost je stále využívána. Slouží k rozpoznávání izolovaných slov (signálů) nebo krátkých úseků. Tento algoritmus se používá zejména pro rozpoznávání řeči a izolovaných, nebo klíčových slov. Jako vzor pro rozpoznávání se používá ve většině případů referenční nahrávka. Postupně, pomocí algoritmu, lze porovnat řečníkem vyslovené slovo (úsek slov) s danou referenční nahrávkou a spočítat vzdálenost cesty DTW. Hledá se ten referenční signál, který měl s daným testovaným signálem nejmenší vzdálenost.

Každý signál je vyjádřen posloupností vektorů. Jako testovaný signál budeme považovat signál **A**:

$$\mathbf{A} = \{\mathbf{a}(1), \mathbf{a}(2), \mathbf{a}(3), \dots, \mathbf{a}(n), \dots, \mathbf{a}(I)\}, \quad (1.3.)$$

a jako referenční signál pod písmenem **B**:

$$\mathbf{B} = \{\mathbf{b}(1), \mathbf{b}(2), \mathbf{b}(3), \dots, \mathbf{b}(m), \dots, \mathbf{b}(J)\}. \quad (1.4.)$$

Znamená to tedy, že  $\mathbf{a}(n)$  značí  $n$ -tý vektor testovaného signálu **A** a  $\mathbf{b}(m)$  je naopak  $m$ -tý vektor referenčního signálu **B**. DTW hledá v rovině  $(n, m)$  optimální cestu

$$m = \Psi(n). \quad (1.5.)$$

Tato cesta minimalizuje funkci  $D$ , což je vzdálenost mezi jednotlivými vektory signálů **A** a **B**:

$$D(\mathbf{A}, \mathbf{B}) = \sum_{n=1}^I d[\mathbf{a}(n), \mathbf{b}(\Psi(n))], \quad (1.6.)$$

kde  $d[\mathbf{a}(n), \mathbf{b}(\Psi(n))]$  je lokální vzdálenost mezi  $n$ -tým vektorem testovaného slova a  $m$ -tým vektorem referenčního slova. Je zavedena časová proměnná  $k$ . Časové proměnné  $m$  a  $n$  jsou takovými funkcemi  $k$ , že platí:

$$\begin{aligned} n &= i(k), \\ m &= j(k). \end{aligned} \quad (1.7.)$$

V případě, že jsou dva signály, u kterých známe počáteční a koncové body, můžeme vyjádřit omezení funkci DTW hraničními body,

$$\begin{aligned} i(1) &= 1, & j(1) &= 1, \\ i(K) &= I, & j(K) &= J. \end{aligned} \quad (1.8.)$$

Váhová funkce  $\widehat{W}$  závisí na lokální cestě, existuje pět základních typů váhové funkce [34] a zde je zobrazena jedna z nich, která byla použita:

typ a) symetrická váhová funkce

$$\widehat{W} = [i(k) - i(k-1)] + [j(k) - j(k-1)], \quad (1.9.)$$

kde  $i(0) = j(0) = 0$ .

Jednotlivé prvky matice **G** jsou rekurzivně vypočteny dle vztahu:

$$g[i(k), j(k)] = \min_{\{i(k), j(k)\}} \{g[i(k-1), j(k-1)] + d[i(k), j(k)]\widehat{W}(k)\}, \quad (1.10.)$$

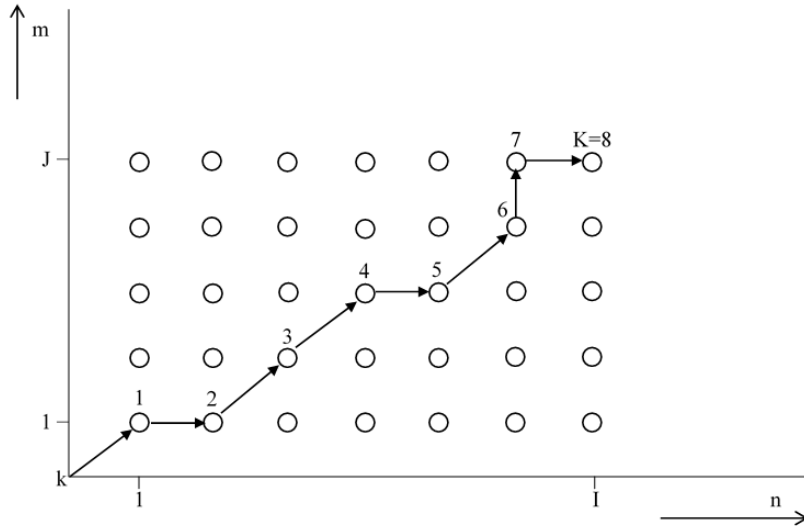
kde  $k = 1, \dots, K$  a  $d[i(k), j(k)]$  je příslušná lokální vzdálenost z matice  $\mathbf{D}$ . Je-li do vztahu (1.12) dosazena váhová funkce typu a), pak lze odvodit konkrétní vztah pro výpočet lokálního omezení typu I [34], který je dán vztahem:

$$g(n, m) = \min \begin{cases} g(n, m-1) + d(n, m), \\ g(n-1, m-1) + 2d(n, m), \\ g(n-1, m) + d(n, m). \end{cases} \quad (1.11.)$$

Konečnou normalizovanou vzdálenost mezi obrazy  $\mathbf{A}$  a  $\mathbf{B}$  lze vyčíslit dle vztahu:

$$D(\mathbf{A}, \mathbf{B}) = [N(\hat{W})]^{-1} g[i(K), j(K)] = [N(\hat{W})]^{-1} g[I, J]. \quad (1.12.)$$

Bude-li hodnota  $\mathbf{D}$  rovná 0 znamená to, že obrazy  $\mathbf{A}$  a  $\mathbf{B}$  mají největší shodu, takže jsou si nejvíce podobné, v opačném případě pak bude mít hodnotu 1, více v [9], [33], [34]. Na Obr. 6 je zobrazena cesta funkce DTW.

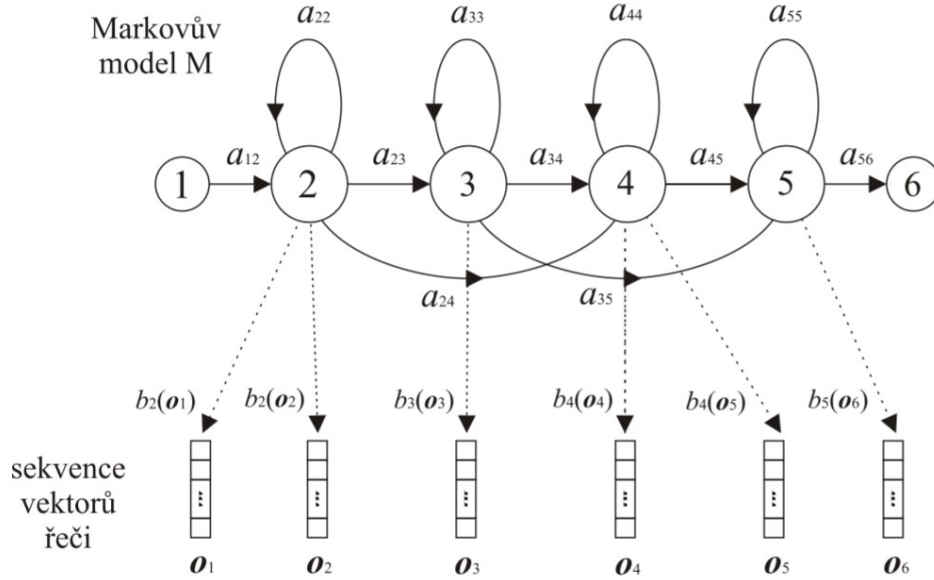


Obr. 6: Funkce DTW pro testovaný a referenční obraz v rovině  $(n, m)$ .

## 2.4 Skryté Markovovy modely HMM

Skryté Markovovy modely jsou statickým nástrojem pro modelování náhodných procesů. Jedná se o modely, jejichž funkce je založena na pravděpodobnostním automatu. V každém časovém okamžiku  $t$  změní svůj stav podle daných pravděpodobností přechodu do stavu  $s_j$ . Při každé změně je generován vektor pozorování  $\mathbf{o}_t$ , který má výstupní pravděpodobnost  $b_j(\mathbf{o}_t)$ . Na následujícím Obr. 7 je zobrazen Markovův model se šesti stavy [35]. Tento model se skládá ze čtyř emitujících stavů a dvou neemitujících stavů. Neemitující stavy negenerují při přechodu do jiného stavu vektor pozorování, slouží pro navazování modelů, při rozpoznávání spojitě řeči. Mezi všemi navazujícími stavy se nacházejí přechodové pravděpodobnosti  $a_{ij}$ . Tyto přechodové pravděpodobnosti určují

pravděpodobnost při přechodu ze stavu  $s_i$  do  $s_j$ . Ke každému stavu jsou přiřazeny funkce rozdělení výstupní pravděpodobnosti  $b_j(\mathbf{o}_t)$ .



Obr. 7: Markovův model se šesti stavy.

### 2.4.1 Rozpoznávání izolovaných slov

Mějme vektory pozorování, které jsou parametrické vektory vypočítané pro každý rámec řečového signálu. Potom každé slovo bude popsáno posloupností těchto vektorů:

$$\mathbf{O} = [\mathbf{o}_1, \mathbf{o}_1, \dots, \mathbf{o}_T]. \quad (1.13.)$$

Symbol  $T$  vyjadřuje celkový počet vektorů pozorování neboli počet časových okamžiků, při kterých dochází ke generování vektorů pozorování.

Dále máme k dispozici slovník  $N$  se slovy  $w_i$ . Problém rozpoznávání izolovaných slov pak můžeme popsat pomocí podmíněné pravděpodobnosti slova  $w_i$ , známe-li vektor  $\mathbf{O}$ :

$$\arg \max_i \{P(w_i|\mathbf{O})\}, \quad (1.14.)$$

kde  $w_i$  je  $i$ -té slovo ze slovníku.

Pravděpodobnost  $P(w_i|\mathbf{O})$  nelze vypočítat přímo, ale lze ji vypočítat pomocí Bayesova vztahu [34], [35]:

$$P(w_i|\mathbf{O}) = \frac{P(w_i|\mathbf{O})P(w_i)}{P(\mathbf{O})}, \quad (1.15.)$$

kde  $P(w_i)$  je apriorní pravděpodobnost výskytu  $w_i$ .  $P(\mathbf{O})$  je pravděpodobnost posloupnosti vektorů pozorování a  $P(w_i|\mathbf{O})$  je podmíněná pravděpodobnost posloupnosti  $\mathbf{O}$ .

Z předchozího obrázku, na kterém se nachází šestistavový model je sestavena sekvence:

$$X = 1,2,2,3,4,4,5,6, \quad (1.16.)$$

ale první a poslední stav je neemitující. Pravděpodobnost posloupnosti  $\mathbf{O}$  je generována model  $M$ , pohybující se přes stavovou sekvenci  $X$ . Takže platí:

$$P(\mathbf{O}, X|M) = a_{12}b_2(\mathbf{o}_1)a_{22}b_2(\mathbf{o}_2)a_{23}b_3(\mathbf{o}_3) \dots \quad (1.17.)$$

V praxi je však známá pouze pravděpodobnost posloupnosti  $\mathbf{O}$  a stavová sekvence  $X$  je skrytá. To je důvod proč se tato metoda nazývá skryté Markovovy modely.

Vzhledem k tomu, že neznáme stavovou sekvenci, pravděpodobnost posloupnosti se vypočte součtem všech stavových sekvencí  $X = x(1), x(2), \dots x(T)$ :

$$P(\mathbf{O}|M) = \sum_x a_{x(0)x(1)} \prod_{t=1}^T b_{x(t)}(\mathbf{o}_t) a_{x(t)x(t+1)}, \quad (1.18.)$$

kde  $x(0)$  je vstupní stav modelu a  $x(T+1)$  je omezen jako výstupní stav modelu.



### 3 Potlačení šumu

Řečové signály lze v praxi jenom zřídka zaznamenávat v čisté formě a jsou téměř vždy kontaminovány šumem okolí, který pochází z různých zdrojů. Právě tato kontaminace hlukem způsobuje velkou chybu a změnu charakteristik řečových signálů a tím narušuje komunikaci na úrovni člověk-člověk nebo člověk-stroj. Z tohoto důsledku jsou vyvinuty a stále vyvíjeny techniky digitálního zpracování. Tyto techniky vyfiltrují řeč kontaminovanou hlukem okolí a dostaneme tím čistý řečový signál, který slouží dalšímu zpracování.

Signál kontaminovaný šumem  $y(m)$  můžeme napsat jako

$$y(m) = x(m) + n(m), \quad (1.19.)$$

kde  $x(m)$  představuje čistý řečový signál a  $n(m)$  je šum.

Metody, které slouží pro zkvalitnění řeči jsou založeny na počtu mikrofónů používaných pro sběr dat. Dají se rozdělit na jednokanálové, dvoukanálové nebo vícekanálové. Ačkoli jsou vícekanálové metody lepší než jednokanálové, stále se používají i jednokanálové metody z důvodu jednoduché implementace a snadnosti výpočtů [36].

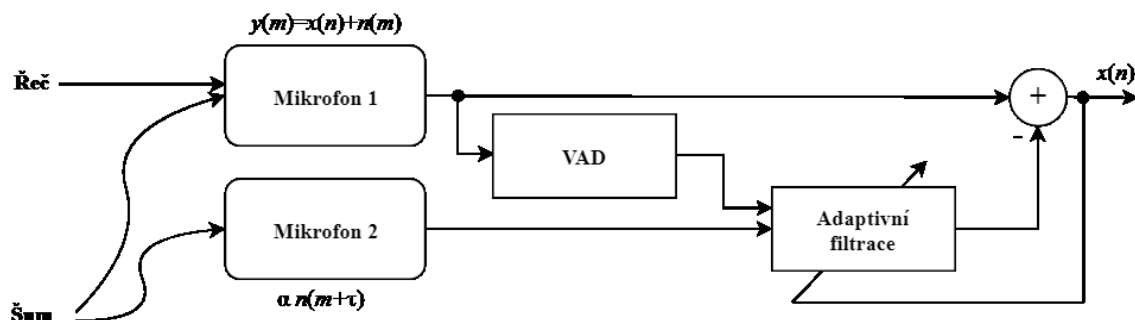
#### Jednokanálové metody

Jednokanálové metody používají pouze jeden mikrofón pro zaznamenávání řeči kontaminované hlukem okolí. Informace o čisté řeči a hluku nejsou k dispozici [36]. Většina jednokanálových algoritmů potřebuje znát odhad spektra hluku. To se obvykle provádí pomocí detekce řeči a pauzy a vyhodnocováním segmentů hluku pomocí VAD [37]. V praxi je tento úkol obtížný z důvodu výskytu nestacionárního hluku. Nevýhodou je potřeba relativně dlouhých minulých segmentů hlukem kontaminované řeči.

#### Vícekanálové metody

Vícekanálové metody využívají k měření dvou a více mikrofónů, které slouží k měření referenčních hodnot šumu. Při zohlednění prostorových vlastností zdroje šumu, lze lépe řešit omezení jednokanálových metod, v přítomnosti nestacionárního šumu [37], [38].

Při komunikaci pomocí komunikačních zařízení například v kokpitu letadla, může být použit směrový mikrofón. Tento mikrofón bude sloužit pro získání okamžité amplitudy šumu. Tím dokážeme získat signál  $x(m)$  odečtením odhadnutého šumu ze signálu, který je tímto šumem kontaminován. Na Obr. 8 lze vidět systém se dvěma měřicími mikrofony, kdy jeden mikrofón zaznamenává řeč kontaminovanou šumem. Druhý mikrofón, který je umístěn v určité vzdálenosti od prvního mikrofónu, zaznamenává hluk. Faktor útlumu  $\alpha$  a časové zpoždění  $\tau$ , představují zjednodušený model simulace mikrofónu umístěného v prostoru. Tento systém pracuje tak, že hluk z jednoho mikrofónu je zpracováván pomocí adaptivního filtru a tento hluk kontaminuje signál řeči. Adaptivní potlačení šumu je účinnější při potlačení nízkofrekvenční části šumu, ale obecně trpí nestacionárním charakterem signálu [31], [39].

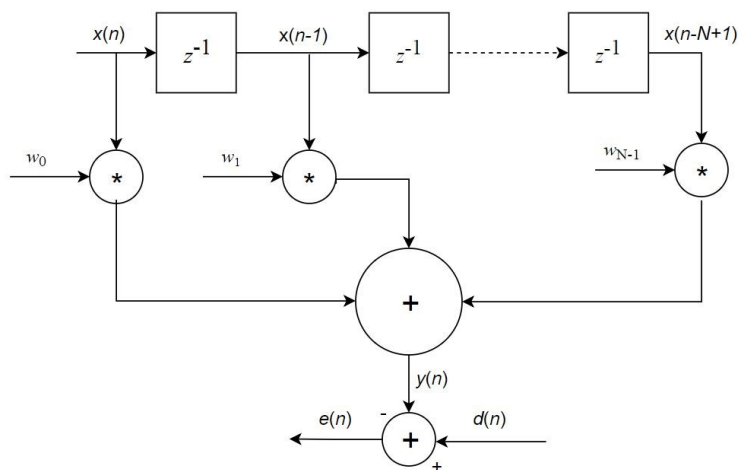


Obr. 8: Konfigurace vícekanálového systému.

### 3.1 Adaptivní filtrace

V reálných aplikacích je potřeba realizovat adaptivní filtrační algoritmy na hardwarových zařízeních. Těmto zařízením se musí dát jasný popis standardního algoritmu LMS. Motivem adaptivního filtrování je obnovení signálu nesoucího informace ze zašuměného signálu. Lze je rozdělit do dvou kategorií. Neadaptivní a adaptivní filtrování. Adaptivní metody filtrování jsou nejvíce používané metody pro kontrolu výkonu a chování řečových signálů [40].

Adaptivní filtrační proces nabízí existenci společného faktoru mezi konvergenční rychlostí a chybovým signálem, pomocí velikosti kroku  $\mu$ . Obr. 9 zobrazuje základní blokový diagram adaptivního filtru [41].



Obr. 9: Blokový diagram adaptivního filtru.

Vstupní vektor  $x(n)$  představuje vektor vzorků,  $y(n)$  označuje výstup adaptivního filtru,  $d(n)$  představuje požadovaný výstupní signál,  $e(n)$  označuje chybový signál. Pro výslednou aplikaci je výstupní veličinou spíše chybový signál regulačního obvodu, který se získá pomocí odečtení výstupního signálu od signálu reálného. Parametry  $w_i$  představují koeficienty vektoru vah transversálního FIR filtru,  $z^{-1}$  reprezentuje zpoždění.

Vektor vah transversálního filtru má podobu:

$$\mathbf{w}(n) = [w_0(n), w_1(n), w_2(n), \dots, w_{N-1}(n)]^T. \quad (1.20.)$$

Počet prvků tohoto vektoru odpovídá řádu FIR filtru  $N$ .

Výstupní signál adaptivního filtru lze zapsat pomocí rovnice:

$$y(n) = \sum_{i=0}^{N-1} w_i(n)x(n-i). \quad (1.21.)$$

Jedná se o filtr  $N$ -tého řádu, proto je rozsah indexace od 0 do  $N - 1$ . Předchozí rovnice může být také vyjádřena pomocí skalárního součinu impulzní odezvy vektoru a vstupního vektoru:

$$y(n) = \mathbf{w}(n) \cdot \mathbf{x}(n). \quad (1.22.)$$

Cílem celého procesu adaptace vah je snižování účelové funkce  $\xi(n)$  na minimum. Tato hodnota je závislá pouze na chybové funkci:

$$\xi(n) = d(n) - y(n). \quad (1.23.)$$

Pro nalezení těchto parametrů se řeší optimalizace chybové funkce, která může být podmíněna deterministicky nebo statisticky. Příkladem statistického pojetí je algoritmus LMS, pro deterministický návrh je užíván algoritmus RLS [41].

### 3.1.1 Adaptivní algoritmus LMS

Least Mean Square LMS neboli algoritmus metody nejmenších čtverců. Jedná se nejpoužívanější adaptivní algoritmus, jelikož je výpočetně nejjednodušší z hlediska náročnosti na paměť. To představuje jeho hlavní výhodu. Tento algoritmus je jednoduchý a velmi snadno realizovatelný. Algoritmus LMS odpovídajícím způsobem aktualizuje velikost kroku vzhledem k hodnotě chybového signálu. Cílem procesu u tohoto algoritmu je postupné snižování účelové funkce  $\xi(n)$  až na její minimum:

$$\xi(n) = E[e^2(n)] = E[(d(n) - y(n))^2]. \quad (1.24.)$$

Vyrovňovací rovnice je vyjádřena:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e(n)\mathbf{x}(n), \quad (1.25.)$$

kde  $x(n)$  je vstupní signál filtru,  $e(n)$  je chybový signál,  $\mu$  je velikost kroku a  $\mathbf{w}(n)$  je vektor váhy. Velikost kroku má vliv na rychlost a stabilitu konvergence u tohoto algoritmu. Jedná se o malou kladnou konstantu. Bude-li tato hodnota příliš malá, bude adaptace na signál příliš pomalá. Naopak, bude-li tato konstanta příliš velká bude algoritmus vykazovat odchylky [41], [42].

Algoritmus LMS se chová následovně:

- Vyhlazuje vstupní signál  $x(n)$  pomocí váhy  $w(n)$  aby získal výstupní signál  $y(n)$ .
- Odhaduje hodnotu chyby jednoduchým odečtením výstupního signálu z požadovaného výstupního signálu:

$$e(n) = d(n) - y(n). \quad (1.26.)$$

- Aktualizuje vah filtru pomocí rovnice (1.24.).

### 3.1.2 Adaptivní algoritmus RLS

Tento algoritmus využívá jako kritériální funkci sumu váhovaných čtverců chybového signálu. Algoritmus RLS má poměrně velkou míru konvergence k optimálním koeficientům. To je užitečné v aplikacích používaných pro zpracování řeči, kde je nutné reagovat na rychlé změny signálu v čase [31], [41], [43].

Struktura filtru zůstává stejná jako u LMS algoritmu, jen má odlišný adaptivní proces vzhledem k použití průměrů [41]. RLS algoritmy se vyznačují vynikajícími výkony při práci v časově proměnných prostředích a extrémně rychlou konvergencí, ale to má za následek, že jsou tyto algoritmy výpočetně náročnější než LMS [41].

Implementace RLS algoritmu je provedena v následujících pěti krocích:

- Výstup filtru je vypočten pomocí vah filtru z předchozí iterace a vstupního vektoru:

$$\bar{y}_{n-1}(n) = \bar{\mathbf{w}}^T(n-1)\mathbf{x}(n). \quad (1.27.)$$

- Vektor středního zesílení je vypočten pomocí rovnice:

$$\begin{aligned} \mathbf{u}(n) &= \tilde{\Psi}_\lambda^{-1}(n-1)\mathbf{x}(n), \\ \mathbf{k}(n) &= \frac{1}{\lambda + \mathbf{x}^T(n)\mathbf{u}(n)}\mathbf{u}(n). \end{aligned} \quad (1.28.)$$

- Hodnota odhadované chyby:

$$\bar{e}_{n-1}(n) = d(n) - \bar{y}_{n-1}(n). \quad (1.29.)$$

- Vektor vah filtru je aktualizován pomocí rovnice:

$$\bar{\mathbf{w}}(n) = \bar{\mathbf{w}}^T(n-1) + \mathbf{k}(n)\bar{e}_{n-1}(n). \quad (1.30.)$$

- Inverzní matice je vypočtena pomocí:

$$\tilde{\Psi}_\lambda^{-1}(n) = \lambda^{-1}(\tilde{\Psi}_\lambda^{-1}(n-1) - \mathbf{k}(n))[\mathbf{x}^T(n)\tilde{\Psi}_\lambda^{-1}(n-1)]. \quad (1.31.)$$

kde  $\mathbf{k}(n)$  je ziskový vektor,  $\lambda$  je faktor zapomínání, který se volí v rozmezí  $<0, 1>$ . V praxi se tato hodnota volí mezi 0,98 až 1, viz. [31].

### 3.2 Slepá separace zdrojů BSS

Tato metoda, je v současné době tématem mnoha výzkumníků v oblasti zpracování signálů a vylepšení řeči [44], [45]. Nejčastěji bývá prováděna pomocí metody ICA, PCA a SVD.

Slepá separace zdrojů se zabývá tím, jak oddělit smíšený signál, když nejsou známy zdrojové signály a přenosové charakteristiky. Od Jie a kolektivu [44] je uvedena konvenční metoda BSS k analýze nestacionárních procesů a je zde ukázán její nedostatek. Proto je v práci uveden zdokonalený algoritmus slepé separace zdrojů založený na analýze nezávislých komponent ICA. Simulace ukazuje, že nová metoda může zlepšit výsledek BSS a časově-frekvenční funkce zdrojového signálu má lepší odraz.

Maticová rovnost je stanovena podle rovnic [46]:

$$\begin{bmatrix} y_1(t) \\ \vdots \\ y_p(t) \end{bmatrix} = \begin{bmatrix} m_{11} & \cdots & m_{1q} \\ \vdots & \ddots & \vdots \\ m_{p1} & \cdots & m_{pq} \end{bmatrix} \cdot \begin{bmatrix} x_1(t) \\ \vdots \\ x_p(t) \end{bmatrix} + \begin{bmatrix} n_1(t) \\ \vdots \\ n_p(t) \end{bmatrix}, \quad (1.32.)$$

$$Y = M \cdot X + N, \quad (1.33.)$$

kde  $n$  je šum přidávaný do signálu. Šum je Gaussovský a vzájemně nezávislý. Cílem BSS je získat  $X$ , což představuje zdroj signálů a  $M$  je lineární transformace a představuje přenos signálu od zdrojů [44], [45].

### 3.3 Analýza nezávislých komponent ICA

Independent Component Analysis (ICA) je metoda separace signálů, která se snaží oddělit od sebe signály, které byly smíchány dohromady. Jedná se například o problém „Cocktail – party problém“, kdy mluví více mluvčích v jeden okamžik a my se snažíme separovat pouze jeden signál od jednoho mluvčího. Budeme-li mít pouze dva mluvčí, kteří vydávají řeč  $s_1(t)$  a  $s_2(t)$ , tak snímače zachycují signály  $x_1(t)$  a  $x_2(t)$  můžeme tento problém popsat rovnicemi [47], [50]:

$$x_1(t) = a_{11}s_1 + a_{12}s_2, \quad (1.34.)$$

$$x_2(t) = a_{21}s_1 + a_{22}s_2.$$

Předpokladem úspěšné separace signálu  $s_1$  a  $s_2$  ze signálu  $x_1$  a  $x_2$  je úvaha, že signály byly statisticky nezávislé a mají negaussovske rozdělení. Směsná matice  $A$  představuje signály prostředí (lineární kombinaci) [50].

Nezávislé komponenty jsou skryté proměnné, které nemohou být přímo pozorovány. Předpokládáme transformaci:

$$\mathbf{x} = \mathbf{A}_{mix}\mathbf{s}. \quad (1.35.)$$

A také, že směšovací matice je čtvercová a po odhadnutí matice  $\mathbf{A}_{mix}$  můžeme počítat s její inverzní maticí  $\mathbf{W}$ . Tím získáme nezávislé komponenty podle rovnice:

$$\mathbf{s} = \mathbf{W}\mathbf{x}. \quad (1.36.)$$

Metoda ICA je velice těsně spojena s metodou, která se nazývá slepá separace zdrojů (BSS). Nalezení separační matice  $\mathbf{S}$  při dodržení předpokladů [48]:

- Počet senzorů je větší než nebo roven počtu zdrojů.
- Zdrojové signály  $s(k)$  jsou v každém časovém okamžiku vzájemně nezávislé.
- Maximálně jeden zdrojový signál může být náhodný vektor s normálním rozdělením.
- Přípustné jsou pouze signály bez aditivního šumu nebo degradované pouze nízkou úrovní aditivního šumu.

Musí však být dodrženy dvě omezení, která jsou základními vlastnostmi BSS (ICA). Prvním omezením je permutace indexů separovaných zdrojů  $\mathbf{y}(k)$ . Při separaci se může stát, že dojde k záměně indexů a separované zdroje nezískáme ve stejném pořadí jako zdrojové. Druhým omezením je, že se nepodaří obnovit separované signály s amplitudou signálů zdrojových.

Předpokládáme-li nezávislé zdroje a nesesingulární směšnou matici  $\mathbf{A}$  pak lze najít separační matici  $\mathbf{S}$ , která se liší podle vztahu:

$$\mathbf{S} = \mathbf{DPA}^{-1}, \quad (1.37.)$$

kde  $\mathbf{P}$  je permutační matice z důvodu permutace indexů,  $\mathbf{D}$  je nesesingulární diagonální matice ovlivňující amplitudy výstupních signálů.

Vztah mezi separovanými zdroji a původními zdroji je pak dán rovnicí:

$$\mathbf{y}(k) = \mathbf{DPs}(k). \quad (1.38.)$$

Amplitudu původních zdrojů se zcela přesně nepovede obnovit, ale povede se obnovit tvar a průběh signálů pomocí separačních algoritmů, což je v praxi důležité [49], [50].

### 3.3.1 Předzpracování ICA

Následující kapitola zobrazuje předzpracování signálu, které je nutné provést, před použitím ICA algoritmu [50]. Jedná se o centrování a bělení.

#### Centrování

Nejzákladnějším a nezbytným předzpracováním je centrování. Vytváří se odečtením středního vektoru  $\mathbf{m} = E\{\mathbf{x}\}$  od  $\mathbf{x}$ . Tím se vytvoří  $\mathbf{x}$  s nulovou střední hodnotou. To implikuje  $\mathbf{s}$  s nulovou střední hodnotou, to lze vidět na základě očekávání na obou stranách rovnice (1.39). Tento druh předzpracování se provádí výhradně pro zjednodušení algoritmu ICA. Po odhadnutí směšovací matice  $\mathbf{A}_{mix}$  s vycentrovanými daty je možné odhad dokončit přidáním střední hodnoty vektoru  $\mathbf{s}$  zpět do centrovaných odhadů  $\mathbf{s}$ . Střední vektor  $\mathbf{s}$  je dán pomocí  $\mathbf{A}_{mix}^{-1}\mathbf{m}$ , kde  $\mathbf{m}$  je střední hodnota, která byla odečtena při předzpracování [50].

#### Bělení

Dalším užitečným předzpracováním je bělení. Před použitím algoritmu ICA a po provedení centrování se transformuje pozorovaný vektor  $\mathbf{x}$  lineárně a tím dostaneme vektor  $\tilde{\mathbf{x}}$ . Tento nový vektor je vybělený, jeho komponenty jsou nekorelované a jejich rozptyl je roven jedné. Kovarianční matice  $\tilde{\mathbf{x}}$  je rovna identifikační matici [50]:

$$E\{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T\} = \mathbf{I}. \quad (1.39.)$$

Jedna populární metoda bělení je použití rozkladu vlastní hodnoty rozkladu (EVD) kovarianční matice.

$$E\{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T\} = \mathbf{E}\mathbf{D}\mathbf{E}^T, \quad (1.40.)$$

kde  $\mathbf{E}$  je ortogonální matice vlastních vektorů  $E\{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T\}$ .  $\mathbf{D}$  je diagonální matice vlastních hodnot  $\mathbf{D} = \text{diag}(d_1, \dots, d_n)$ .  $E\{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T\}$  může být získána z dostupných vzorků  $\mathbf{x}_1, \dots, \mathbf{x}(T)$ . Bělení může být provedeno pomocí rovnice [50]:

$$\tilde{\mathbf{x}} = \mathbf{E}\mathbf{D}^{-1/2}\mathbf{E}^T\mathbf{x}, \quad (1.41.)$$

kde  $\mathbf{D}^{-1/2}$  je vypočteno jako  $\mathbf{D}^{-1/2} = \text{diag}(d_1^{-1/2}, \dots, d_n^{-1/2})$ . Poté je lehké ověřit rovnici (1.43). Bělicí transformaci směšovací matice získáme novou matici  $\tilde{\mathbf{A}}$  podle rovnice:

$$\tilde{\mathbf{x}} = \mathbf{E}\mathbf{D}^{-1/2}\mathbf{E}^T\mathbf{A}_{mix}\mathbf{s} = \tilde{\mathbf{A}}\mathbf{s}. \quad (1.42.)$$

Užitečnost bělení je hlavně v tom, že nová směšovací matice je ortogonální. Pomocí bělení lze tedy snížit složitost algoritmu ICA [50].

### 3.4 Analýza hlavních komponent PCA

Principal Component Analysis PCA využívá podobný model jako ICA bez vektoru aditivních šumů. Tato metoda je příbuzná rozkladu pomocí singulárních čísel a bývá označována jako Hotellingova transformace [9], [44]. Cílem této metody je nahradit původní proměnné, které jsou korelované, hlavními komponentami, které jsou nekorelované. Tyto proměnné jsou ve většině případů lineární kombinací původních proměnných a obsahují o nich co nejvíce informací. Hlavním účelem PCA je komprese dat a výběr příznaků, což jsou signály nesoucí nejvíce užitečné informace. Vstupem této metody je matice  $\mathbf{X}$ , která obsahuje  $n$  vzorků pro  $p$  hlavních komponent. Předpokládáme, že matice  $\mathbf{X}$  je sloupcově centrovaná a průměry jednotlivých sloupců se rovnají nule. Matice  $\mathbf{Z}$  bude obsahovat sloupce hlavních komponent vytvořené lineární kombinací. Inverzní transformace je dána podle rovnice [52]:

$$\mathbf{Z} = \mathbf{X} \cdot \mathbf{A}, \quad (1.43.)$$

$$\mathbf{X} = \mathbf{Z} \cdot \mathbf{A}^T, \quad (1.44.)$$

kde  $\mathbf{A}$  je ortogonální (nekorelovaná) matice.

Na základě vzájemných transformací lze určit, že  $\mathbf{X} * \mathbf{X}^T = \mathbf{Z} * \mathbf{Z}^T$ . Z této rovnice plyne, že v obou souřadnicových systémech jsou zachovány Eukleidovské vzdálenosti mezi body a velikosti úhlů, které svírají vektory spojující tyto body s počátkem souřadnic. Matice  $\mathbf{A}$  způsobuje pouze rotaci kolem počátku souřadnic. Více o odvození této metody v [52].

Standartní postup PCA se dá rozdělit na [52]:

- Transformace dat.
- Rozklad kovarianční reps. korelační matice.
- Určení počtu významných hlavních komponent.
- Vizuální zobrazení vícerozměrných dat.

#### 3.4.1 Rozklad pomocí singulárního rozkladu SVD

Singular Value Decomposition SVD je velice výkonnou metodou. Jedná se o metodu, která je založena na maticové transformaci jednoho vektorového prostoru v jiný [53]. Dva vektory  $\mathbf{x}$  a  $\mathbf{y}$  jsou ortogonální v tom případě, že platí rovnice  $\mathbf{x}^T \mathbf{y} = 0$ . Matice  $\mathbf{A}$  je čtvercového typu, její sloupce jsou ortogonální vektory, které mají délku jedna. Tato matice obsahuje minimálně tolik řádků, co sloupců  $m \geq n$ .

Rovnice SVD metody se skládá ze součinu tří matic:

$$\mathbf{A} = \mathbf{U} \mathbf{S} \mathbf{V}^T, \quad (1.45.)$$

kde  $\mathbf{U}$  je ortogonální matice o velikosti  $m \times n$ ,  $\mathbf{V}$  je ortogonální matice o velikosti  $n \times n$ . Matice  $\mathbf{S}$  je  $m \times n$  diagonální matice s  $s_{ij} = 0$  pokud  $s_{ii} = s_i \geq 0$ . Množství  $s_i$  se nazývá singulárními



hodnotami matice  $\mathbf{A}$ , sloupce  $\mathbf{U}$  a  $\mathbf{V}$  se nazývají levými a pravými singulárními vektory. Z komponentů SVD můžeme určit mnoho vlastností původní matice [53], [54].

De Lathauwer a kolektiv [54] diskutovali výhody a nevýhody metod používajících SVD. Zjistili, že směšovací matici lze snadno odhadnout, ale na druhé straně je tento algoritmus výpočetně složitý. V současné době jsou tyto metody rozkladu běžně účinnou a velice účinnou metodou pro filtraci šumu.

## 4 Experimentální část

Výsledkem diplomové práce jsou dva programy vytvořené ve vývojovém prostředí LabVIEW. První z programů byl vytvořen z důvodu potřeby nahrávání vlastních nahrávek řeči a hluků. Tyto nahrávky byly pořízeny v prostorech bezdozvukové komory, automobilu, továrny a letadla. Popis jednotlivých měření a rozmístění mikrofونů při měření je zobrazen v následující kapitole.

Druhý program, který byl v rámci diplomové práce zpracován, slouží pro vyhodnocování výsledků filtrace. Podle zpracované literární rešerše se v programu nacházejí dvě metody kompenzace šumu pro rozpoznávač řeči. Funkce programu je popsána v kapitole 4.4.

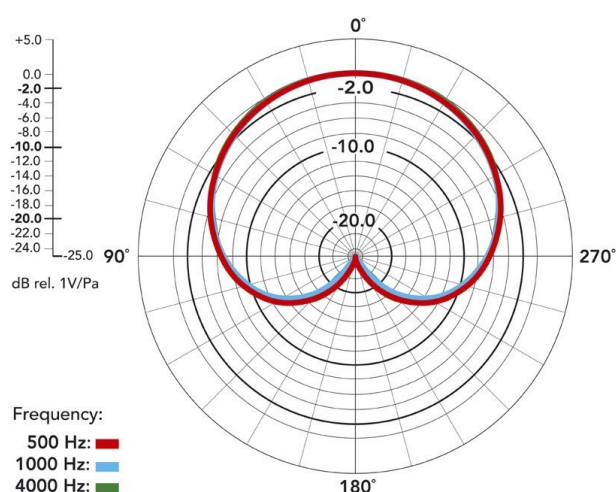
### 4.1 Měření a použitá zařízení

Nahrávání bylo realizováno pomocí notebooku, ke kterému byla připojena zvuková karta Steinberg UR44 (Obr. 10) [55]. Tato zvuková karta obsahuje 24bitové AD a DA převodníky a podporuje vzorkovací frekvenci až 192kHz. Jednou menší nevýhodou této zvukové karty je nutnost použití externího napájení. Dalším vybavením pro měření byly čtyři měřicí mikrofony značky Rode NT5 (Obr. 10) [56]. Jedná se o membránový kondenzátorový mikrofون. Ke své činnosti potřebuje fantomové napájení, které je zajištěno zvukovou kartou. Propojení mikrofонů a zvukové karty bylo provedeno pomocí kabelů Bespeco PYMB.



Obr. 10: Měřicí mikrofون Rode NT5 a zvuková karta Steinberg UR44.

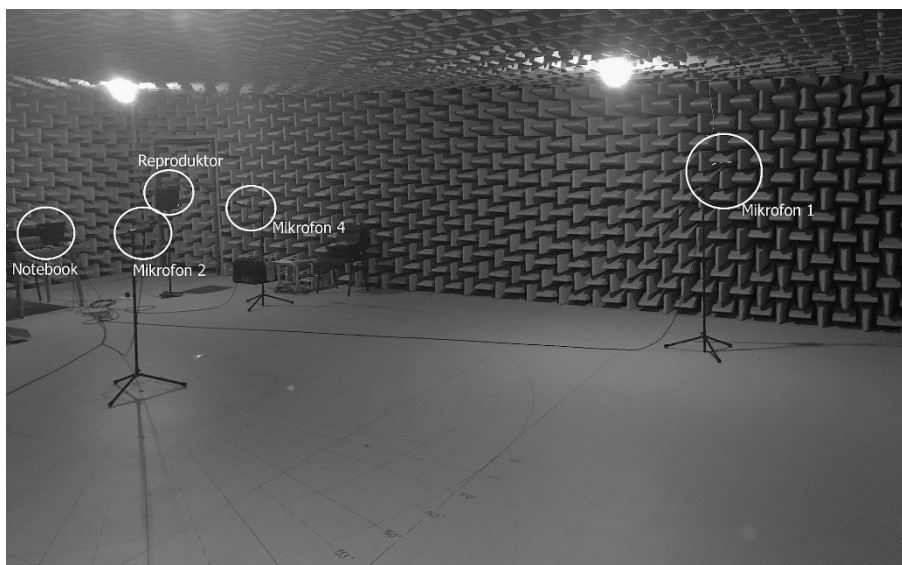
Směrová charakteristika použitých mikrofонů je zobrazena na následujícím Obr. 11. Jedná se o takzvanou kardioidní charakteristiku. Mikrofony s touto charakteristikou nejsou úzce směrové, takže v menší míře zabírají i okolní zvuky (hluk okolí), obrázek je převzatý z [55].



Obr. 11: Směrová charakteristika Rode NT5.

Zvuková karta podporuje vzorkovací frekvenci až 192kHz. Byla tedy možnost si vybrat ideální hodnotu pro uskutečnění měření vybraných nahrávek řeči a hluku. Ze znalosti frekvenčního rozsahu řeči, kdy se jeho nejvýznamnější část pohybuje v rozsahu od 300 Hz do 3000 Hz. Dále ze znalosti Shannon-Kotělnikova teorému, který nám říká, že vzorkovací frekvence musí být rovna minimálně dvojnásobku nejvyšší harmonické složky. Můžeme říct, že ideální vzorkovací frekvence by byla 16 kHz. Ovšem zvuková karta při použití měření se čtyřmi mikrofony nedovoluje nastavení menší vzorkovací frekvence, než je 44100 Hz. Z tohoto důvodu byla tato frekvence použita jako výsledná vzorkovací frekvence při měření.

Měření nahrávek řeči probíhalo v akustické bezdozvukové komoře. Jedná se jen o částečně bezdozvukovou komoru, jelikož se v ní nenachází speciální podlaha. Tato komora se nachází v areálu Vysoké Školy Báňské, Technické Univerzity Ostrava. Tato komora má stejné fyzikální vlastnosti jako volný prostor. Představuje tedy prostředí, ve kterém nevznikají při šíření zvuku nežádoucí vlivy, jako jsou odrazy nebo ohyby zvuku. Základním prvkem bezdozvukové komory jsou klíny z absorpčních materiálů, jako je skelná vata. Klíny jsou opatřeny potahem, který stabilizuje tvar klínu a zabraňuje vypadávání vláken materiálu. Těmito klíny jsou opatřeny stěny a strop komory. Pohlcování zvuku není jedinou funkcí těchto klínů, ale zároveň slouží k izolaci od vnějšího hluku a udržování konstantní teploty uvnitř komory.



Obr. 12: Akustická bezdozvuková komora.

Na Obr. 12 lze vidět rozložení tří měřících mikrofónů. Čtvrtý měřící mikrofón se nachází vlevo v místnosti a na obrázku jej nelze vidět. Dále se na obrázku nachází reproduktor a vlevo i notebook se zvukovou kartou Steinberg UR44. V komoře probíhalo měření nahrávek řeči, konkrétně pomocí mikrofónu, který se nachází na obrázku vpravo. Nahrávky řeči byly měřeny ze vzdálenosti kolem deseti centimetrů. Výška mikrofónu byla nastavena na 170 centimetrů. Ve stejné komoře byly změřeny i hluky, se kterými se můžeme setkat v domácnosti. Konkrétně se jednalo o nahrávky šumu myčky, pračky, ventilátoru. Hluky byly produkovány pomocí reproduktoru.

Další měření probíhalo v osobním automobilu (Obr. 13). Tři ze čtyř měřících mikrofónů byli umístěny v prostoru celého automobilu a nacházeli se na úchytech na stropě. Poslední mikrofón byl umístěn nad zpětným zrcátkem v místech, kde se obvykle takový mikrofón pro hlasové ovládání nachází. Měření v automobilu probíhalo za účelem změření hluků při různých rychlostech a v kombinaci s otevřenými a zavřenými okny.



Obr. 13: Rozložení mikrofónů v elektrickém vozidle.

Další měření bylo uskutečněno v ultralehkém letadle ALTO 912. Kabinu letounu lze vidět na Obr. 14. Dva mikrofony zde byly umístěny v kapsách nacházejících se na bocích letadla (sít'ovaná kapsa). Tyto mikrofony byly od trupu letadla odizolovány pomocí molitanu, aby bylo zamezeno přenášení vibrací, jak je zobrazeno na Obr. 15.



Obr. 14: Ultralehké letadlo ALTO 912.

Třetí mikron byl spolu s notebookem a zvukovou kartou umístěn, za pilotem v malém zavazadlovém prostoru. Čtvrtý mikron měl pilot připevněný ke sluchátkům, který kromě hluku vznikajícím v letadle zaznamenával i řeč pilota o stavu rychlosti, výšky a otáček motoru.



Obr. 15: Uložení mikrofonu.

Jak již bylo dříve řečeno, zvuková karta potřebuje ke své činnosti externí napájení. V osobním automobilu byl tento problém vyřešen pomocí měniče napětí zapojeného do autozásuvky. Stejný problém byl třeba vyřešit i v použitém letounu. Tento problém byl řešen pomocí LiPo baterie, která měla dostatečnou kapacitu pro udržení zvukové karty v provozu po dobu překračující jednu hodinu letu.

Poslední měření probíhalo v továrně na výrobu sedaček do automobilů. V této továrně byly změřeny celkem dva druhy hluků, které se zde primárně vyskytovaly. První hluk, který byl v továrně změřený byl hluk velkého lisovacího stroje. Druhý hluk, byl hluk laserové svařčky. Následující obrázky (Obr. 16, Obr. 17) jsou převzaty z internetu a představují druhy strojů, na kterých bylo prováděno měření.



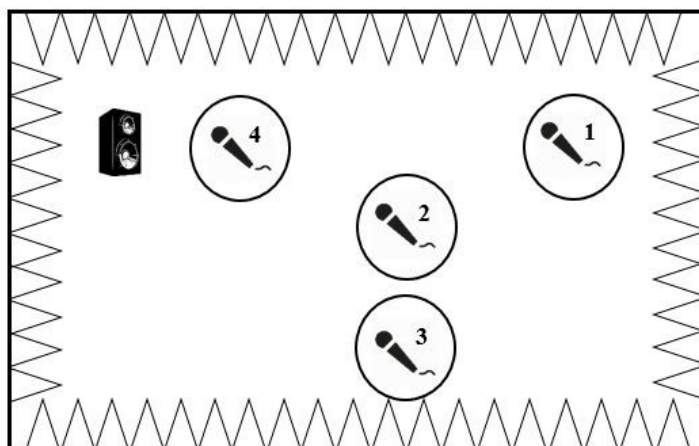
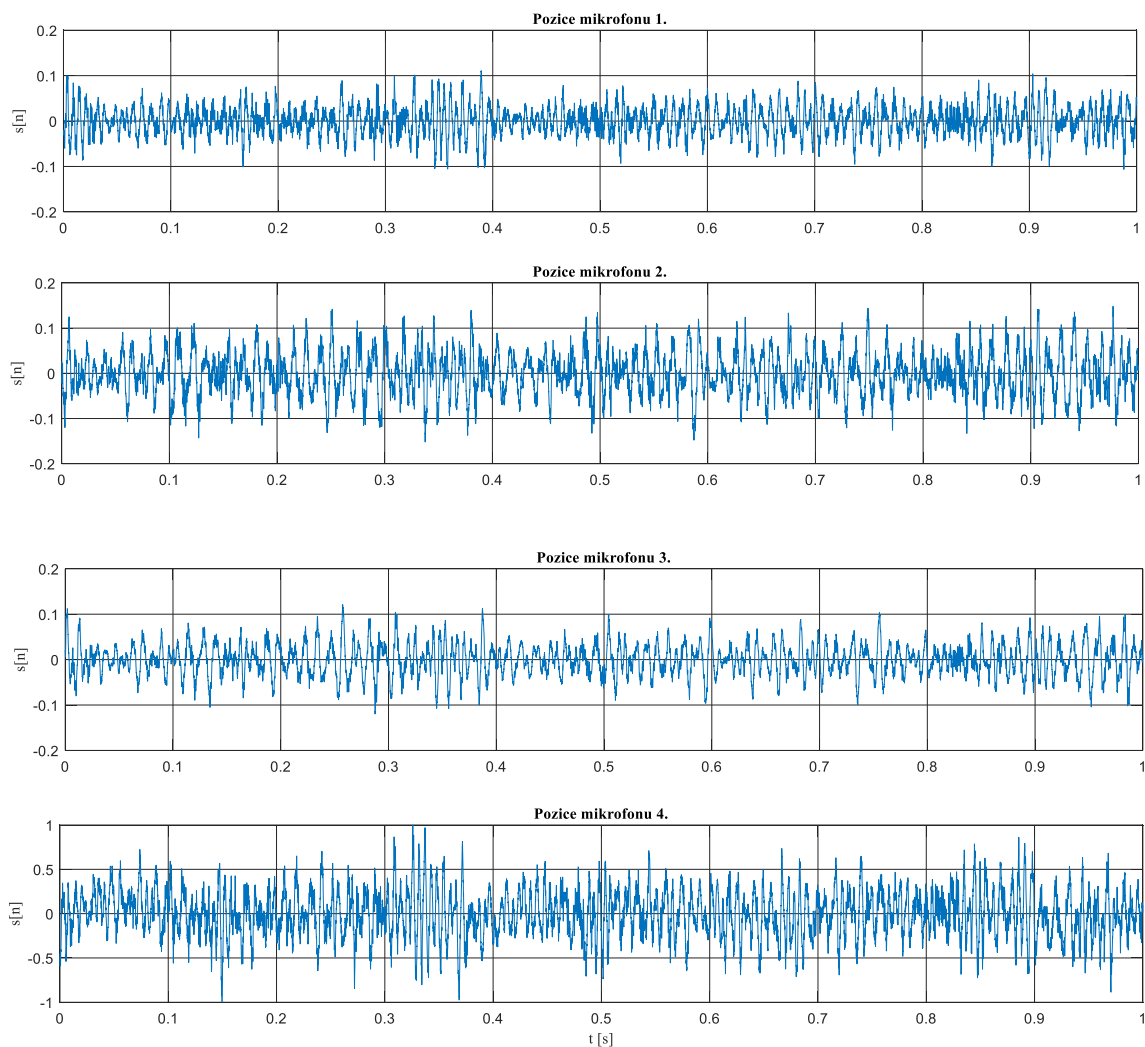
Obr. 16: Hydraulický lisovací stroj.



Obr. 17: Laserové svařovací pracoviště.

V částečně bezdozvukové komoře byly prováděny nahrávky řeči z pozice 1. Nahrávky hluků již nebyly prováděny pouze z jedné pozice, ale byly využity čtyři různá místa pro každý prostor, ve kterém bylo nahrávání prováděno. Pro příklad je následujícím obrázkem zobrazeno rozmístění měřících mikrofónů v bezdozvukové komoře (Obr. 18). Nahrávky řeči byly pořizovány pouze v pozici 1. Při nahrávání hluků, byly využity již všechny pozice. Hluk pořizený z pozice 1 byl v programu přičítán k nahrávce řeči. Takovéto nastavení platí pro všechny prostory, ve kterých bylo pořizováno nahrávání. Tedy na pozici 1 se vždy nacházel mikrofón, který slouží pro hlasové ovládání. V letadle byl tento mikrofón umístěn na sluchátcích pilota. Pro továrnu se mikrofón nacházel vždy u ovládacího panelu daného stroje. U automobilu byl tento mikrofón umístěn nad zpětným zrcátkem, v obvyklém místě jeho výskytu. Poté si uživatel může vybrat ze zbylých pozic, kterou úroveň hluku použije jako referenční pro adaptivní filtraci. Nahrávky hluky byly pořizovány z různých pozic, aby nebyl hluk přičtený k řeči stejný jako jeho referenční úroveň. Na Obr. 18 jsou

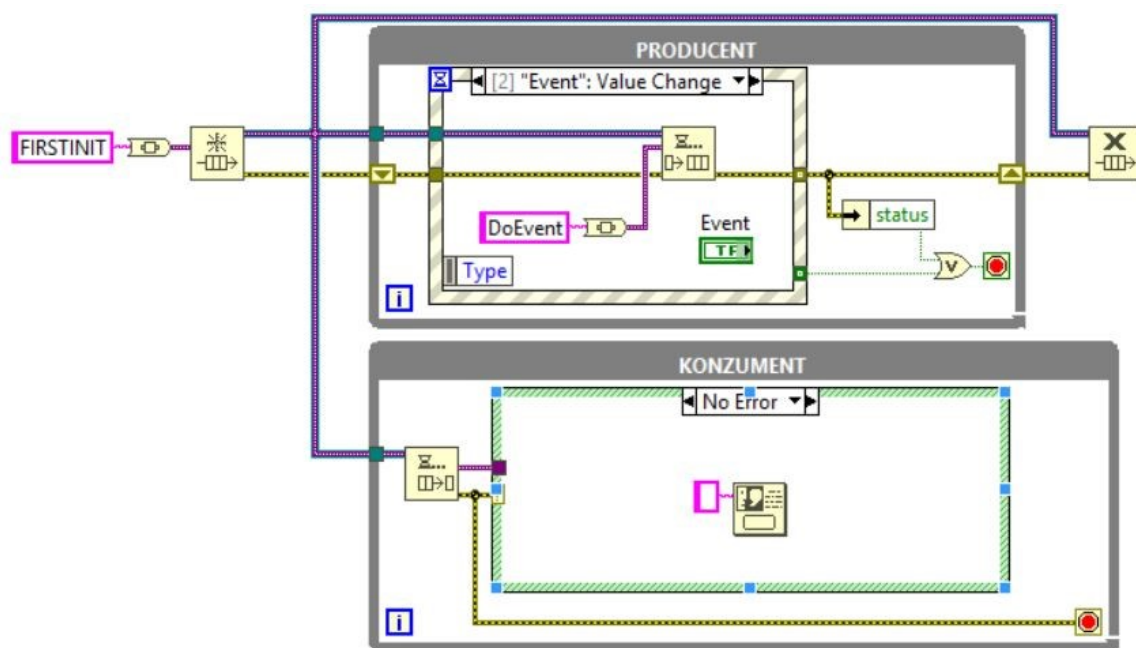
zobrazeny grafy pro hluk pračky z jednotlivých míst, ve kterých měření probíhalo. Lze zde pozorovat, že úroveň hluku je v každé pozici rozdílná z důvodu časového zpoždění a šíření zvuku v prostředí.



Obr. 18: Zobrazení rozmístění mikrofónů v bezdovukové komoře a časové průběhy z jednotlivých pozic při zaznamenávání hluku pračky.

## 4.2 Architektura aplikace

Výsledná aplikace byla vytvořena pomocí struktury producent/konzument, která slouží k lepšímu sdílení dat mezi smyčkami, které běží v různých rychlostech. Celý program je pomocí této struktury zkonstruován. Pracuje na principu, kdy první smyčka produkuje data, jedná se tedy o producenta. Druhá smyčka tyto data vykonává, jedná se tedy o konzumenta. Přenos dat mezi producentem a konzumentem je řešen pomocí front. Fronty tedy slouží k přenosu dat, ale každá fronta se dokáže vázat pouze na jeden konkrétní datový typ. Každá odlišná datová položka musí být přenášena pomocí jiné fronty. Strukturu lze pozorovat na následujícím Obr. 19.



Obr. 19: Struktura producent/konzument.








### 4.3 Rozhraní zvukové karty

Balíček WaveIO se skládá z knihovny a sady rozhraní LabVIEW VIs pro přístup k různým funkcím, který je dostupný online [57]. Pomocí tohoto balíčku můžeme použít ASIO (Audio Stream Input/Output), které je užitečné i pro aplikace využívající vysoký počet kanálů.

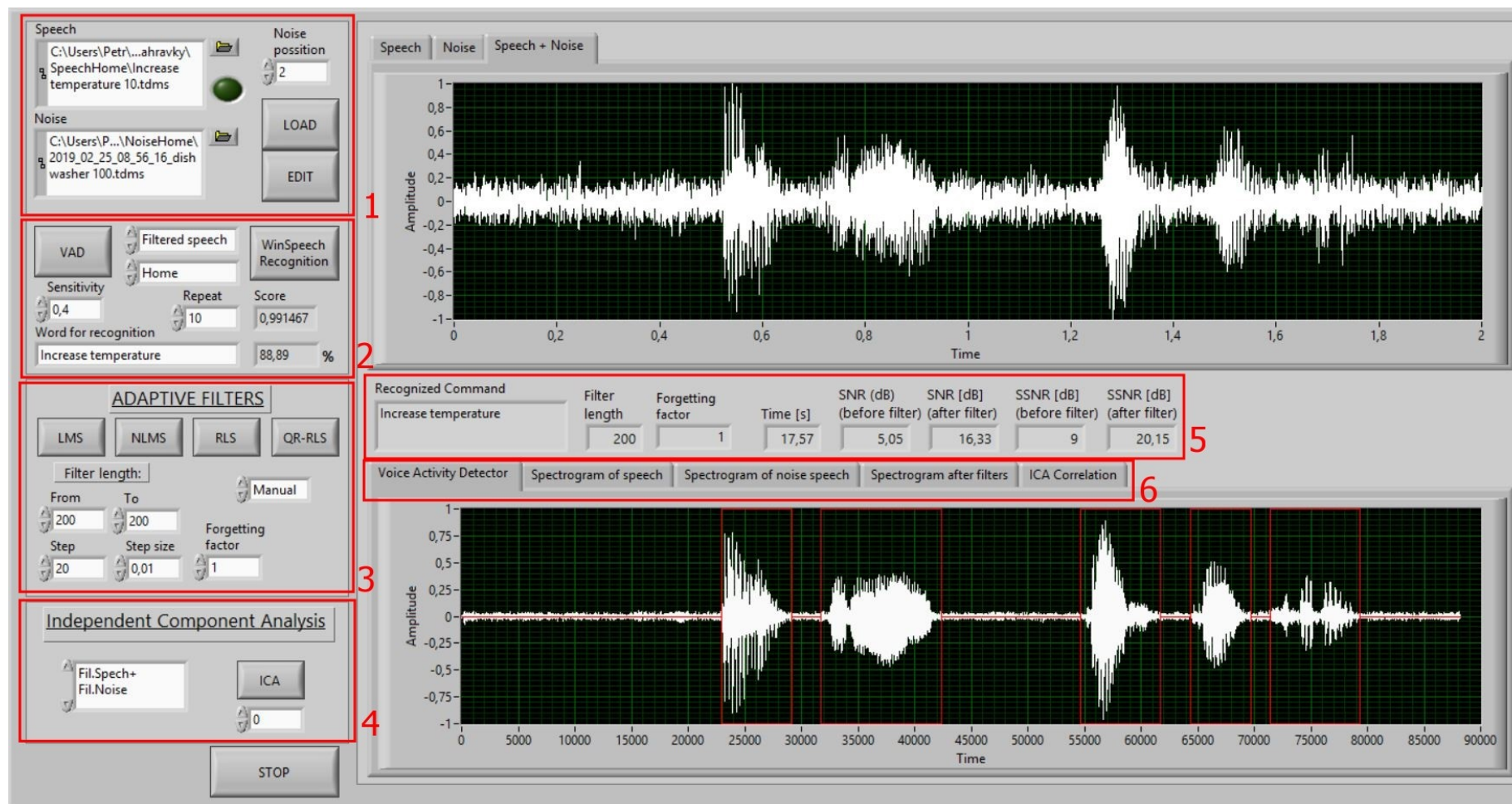
Pro využití tohoto rozhraní je nutné mít zvukovou kartu s ovladačem ASIO. Pro použití ASIO se používá stejné nastavení, jako pro zařízení Windows. Je možné používat pouze jedno zařízení v jednom okamžiku. Toto rozhraní bylo využito zejména z toho důvodu, že bylo potřeba měřit 4 kanály. Při použití tohoto rozhraní bylo i nutné volit vhodnou vzorkovací frekvenci, která byla zvolena na 44100 Hz. Knihovna DLL automaticky vybírá použité rozlišení. Interně pracuje s 32bitovými hodnotami. Proto musí být výstupní data poskytnuta a přijímána s 32bity na vzorek. V Tab. 1 je zobrazena paleta použitých funkcí a jejich význam.

Tab. 1: Paleta dostupných rozhraní (VIs).

Paleta dostupných funkcí rozhraní		
<b>WaveIO_Open</b>	Slouží pro otevírání zvukové karty pro nahrávání nebo přehrávání. Zařízení musí být před vlastním čtením spuštěno. Otevření vrátí chybu, pokud zařízení neexistuje nebo pokud je již otevřeno.	
<b>WaveIO_Start</b>	Spustí referenci zvukové karty.	
<b>WaveIO_Record</b>	Čeká a načítá data ze zvukové karty.	
<b>WaveIO_Stop</b>	Zastaví zvukovou kartu. Je požadováno před zavřením zvukové karty.	
<b>WaveIO_Close</b>	Zavře referenci zvukové karty.	

## 4.4 Popis výsledné aplikace

V následující podkapitole bude popsána výsledná aplikace. Čelní panel této aplikace je zobrazen na následujícím Obr. 20.



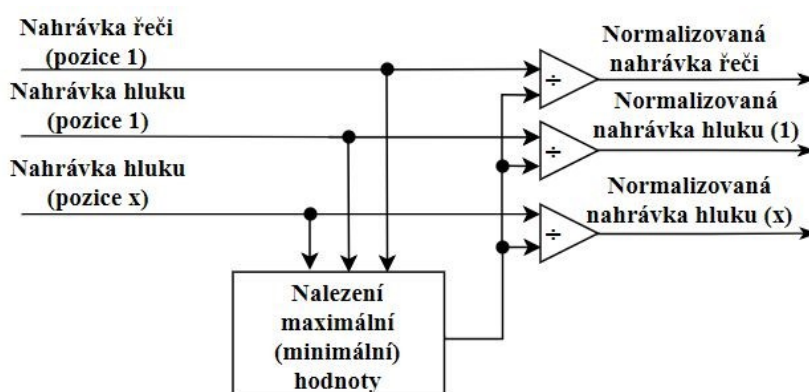
Obr. 20: Čelní panel výsledné aplikace.

Ovládání programu je rozděleno na šest částí:

- 1) První část slouží k nahrání nahrávek řeči a šumu z databáze nahrávek. LED dioda slouží k indikaci chyby, tedy pokud bude některé z polí pro nahrání prázdné a uživatel klikne na tlačítko *LOAD* tak se tato LED dioda zbarví červeně a je tak indikována chyba. Když pomineme případ, že nastala chyba, může po tom, co byly nahrávky úspěšně nahrány do programu, přejít k úpravě pomocí tlačítka *EDIT*. Jelikož jsou nahrávky nahrávány v 32bitovém rozlišení, pohybují se hodnoty amplitudy v milionových hodnotách a pro další zpracování by byly nepoužitelné. Touto úpravou se tedy myslí normalizace všech signálů na rozmezí hodnot od -1 do 1.
- 2) Po tom, co byly vstupní nahrávky upraveny, bude zpřístupněn zbytek čelního panelu. Druhá část čelního panelu slouží k ovládání intenzitního detektoru řeči *VAD*. Průběh této detekce je zobrazen ve spodním grafu a označen červenou křivkou. K detektoru řeči se pojí ještě proměnná *Sensitivity*, pomocí které můžeme určovat citlivost tohoto detektoru. V prvním okně (enumerátoru) pro výběr, si můžeme zvolit, zda se má detekce provádět na odfiltrované řeči (*Filtered speech*), čisté řeči (*Clear speech*) anebo na řeči, která je kontaminována vybraným šumem (*Noisy speech*). Dále se v druhé části vyskytuje rozpoznávač řeči *WinSpeech Recognition* (WSR), který je popsán níže. Proměnná *Repeat* slouží k nastavení počtu opakování pro WSR, kdy se porovnává slovo *Recognized Command* se slovem *Word for recognition*, které si volí uživatel sám podle toho, pro který příkaz se zrovna provádí filtrace. V poslední řadě je zde zobrazena úspěšnost detekce rozpoznávače v procentech a jeho výsledné *Score*.
- 3) Třetí část je spojena s adaptivní filtrací. Uživatel si může volit mezi čtyřmi algoritmy *LMS*, *RLS*, *NLMS* a *QR-RLS*. Dále si uživatel může vybrat, jestli se má filtrace provést automaticky *Auto* anebo si sám vybere hodnoty, které se mají nastavit pro filtraci *Manual*. Pokud si uživatel vybere první metodu, je tento program zkonstruován takovým způsobem, aby pro *LMS* a *NLMS* nastavil postupně délku filtru od hodnoty 10 do 1000 s velikostí kroku 10 a ke všem hodnotám nastavil tři velikosti konvergenční konstanty  $\mu$  (0,001; 0,01; 0,1). Pro každé toto nastavení je vypočítána doba, kterou tato filtrace trvala a hodnota odstupu signálu od šumu SNR. Program bude dále pracovat z tou hodnotou filtrace, která měla nejlepší hodnoty SNR. Pro *RLS* a *QR-RLS* se nastavuje hodnota délky filtru od 20 do 300 s krokem 20 z důvodu velké časové náročnosti a taktéž se zde nastavuje pouze jedna hodnota faktoru zapomínání  $\lambda=1$ .
- 4) Čtvrtá část tohoto programu slouží pro vyhodnocení metody nezávislých komponent *ICA*. Tato metoda je v programu použita, ne za účelem filtrace, ale z důvodu separace dvou signálů od sebe. Kdy tyto signály spolu byly smíchány. Metoda není v programu použita samostatně, ale lze ji použít spolu s adaptivní filtrací. Dále si uživatel může volit ze dvou výstupních signálů. Není totiž přesně dáno, který signál bude na jednotlivých pozicích.

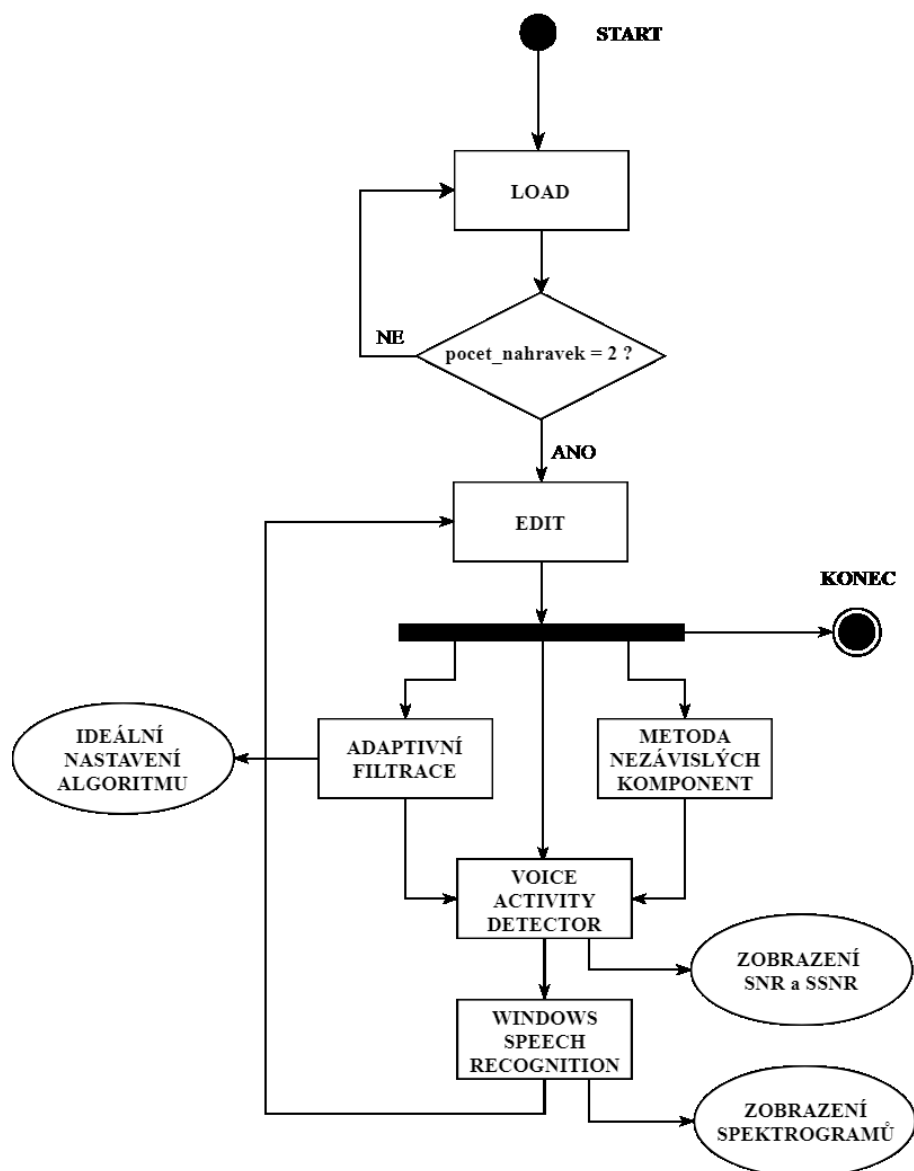
- 5) Pátá a poslední část je spojena spolu se třetí částí a jsou zde zobrazeny výsledky filtrace. Zobrazuje se zde ideální nastavení délky filtru, konvergenční konstanty nebo faktoru zapomínání. Dále je zde zobrazena doba výpočtu *Time* hodnota *SNR* signálu před filtrací, hodnota *SNR* po filtraci a pomocí intenzitního detektoru je zde zobrazena i vypočtená hodnota *SSNR* před a po filtraci. V poslední řadě je zde zobrazena fráze, která byly právě rozpoznána pomocí WSR.
- 6) Poslední část zobrazuje jednotlivé záložky. Mezi těmito záložkami může uživatel přepínat a zobrazovat tím výslednou filtraci v časové doméně. Tak i v časově-frekvenční doméně formou spektrogramu.

Na Obr. 22 je zobrazen stavový diagram výsledné aplikace. Je zde znázorněna funkčnost jednotlivých fází aplikace, tak jak jdou za sebou a jak je možné tento program ovládat. Při spuštění aplikace má uživatel k dispozici pouze tlačítko *LOAD* a musí v prvním kroku provést nahrání řeči a šumu do programu. Jeli nahrání provedeno správně, provede uživatel úpravu nahrávek tlačítkem *EDIT*, kdy se signály normalizují na rozsah hodnot  $\langle -1, 1 \rangle$ . Tato normalizace je provedena ve stejné míře pro všechny nahrávky, aby bylo dodrženo zachování poměrů nahrávek vůči sobě viz. Obr. 21. Ze tří použitých nahrávek se vybere nejvyšší hodnota, nebo nejnižší absolutní hodnota, a původní nahrávky budou pomocí této hodnoty poděleny.



Obr. 21: Zobrazení způsobu normalizace.

Po úspěšné úpravě se uživateli již zpřístupní zbytek programu a může si vybrat ze tří kroků k dalšímu postupu. První krok představuje zvolenou adaptivní filtraci, kdy se vypočítají neoptimalnější parametry filtru podle hodnot *SNR* a tyto hodnoty jsou uživateli vizualizovány na čelním panelu. Poté se provede výpočet *VAD* a následné rozpoznání filtrované řeči pomocí rozpoznávače *WSR*. Posledním krokem je zobrazení spektrogramů pro čistou nahrávku řeči, nahrávku řeči smíchanou se zvoleným hlukem a nahrávku odfiltrované řeči. Druhý krok přistupuje k detekci *VAD* bez použití filtrace. Třetí krok slouží ke kombinaci adaptivní filtrace a metody nezávislých komponent ICA, kdy všechny kroky poté jsou stejné jako při použití první volby.



Obr. 22: Stavový diagram výsledné aplikace.

#### 4.4.1 Realizace detektoru řečové aktivity

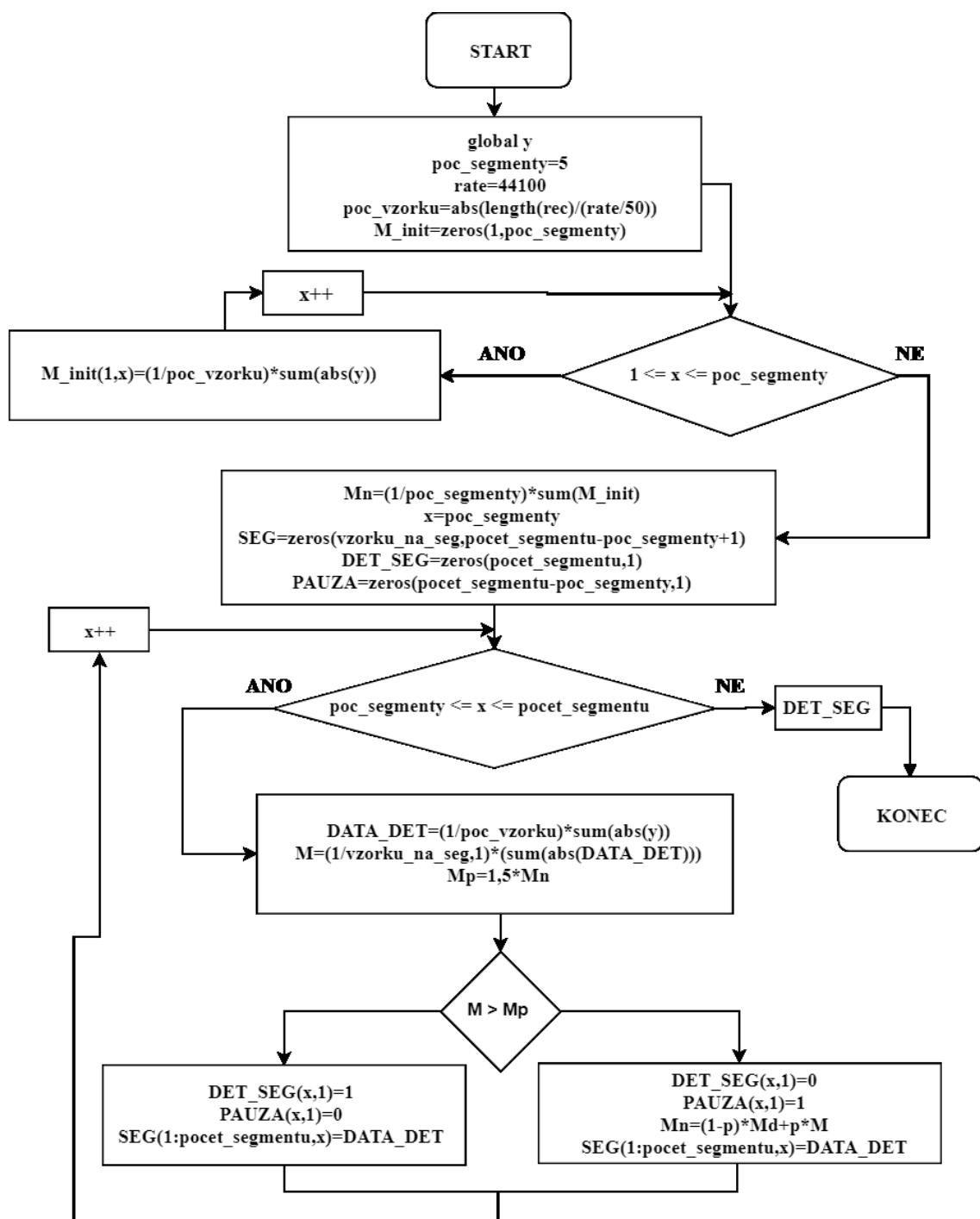
Aby bylo možné vypočítat segmentální hodnotu  $SNR$  ( $SSNR$ ), bylo nutné zpracovat intenzitní detektor řeči VAD. Následující vývojový diagram (Obr. 23) zobrazuje průběh výpočtu detektoru. Před samotným výpočtem se musí pro zvolené nahrávky řeči a šumu provést segmentace. Ideální délka těchto segmentů se pohybuje v rozmezí od 20 ti do 30 ti milisekund. V tomto programu byla délka zvolena na 20 milisekund ( $rate/50$ ). Při vzorkovací frekvenci 44100 Hz, bude délka jednoho segmentu 882 vzorků ( $poc\_vzorku$ ). Pro každý segment je vypočtena hodnota intenzity a uložena do vektoru pro další zpracování ( $M\_init$ ). Po ukončení cyklu je vypočtena průměrná hladina intenzity, přes všechny inicializační segmenty a ve vývojovém diagramu je tato hodnota označena  $Mn$ .

Po provedení inicializační fáze, můžeme přejít k druhému kroku, detekci řeči v signálu. Proměnná  $x$  představuje počet segmentů na začátku nahrávky, ze kterých se vypočítá prahová hodnota intenzity  $Mp$ . Tato hodnota bude sloužit jako detekční práh. Počet počátečních segmentů, pro výslednou aplikaci byl zvolen na hodnotu 5. Po vypočtení prahové hodnoty je prováděno detekování řeči v signálu. Porovnává se zde intenzita testovaného segmentu  $M$  s intenzitou prahovou  $Mp$ . Pro každý segment je vypočtena intenzita  $M$ . Stejně jako v inicializační fázi, jsou zde cyklicky ukládány segmenty intenzity do proměnné  $SEG$ . Cyklus je inicializován pro segmenty náležící intervalu od konce inicializace ( $poc\_segmenty$ ) do celkového počtu segmentů ( $počet\_segmentu$ ).

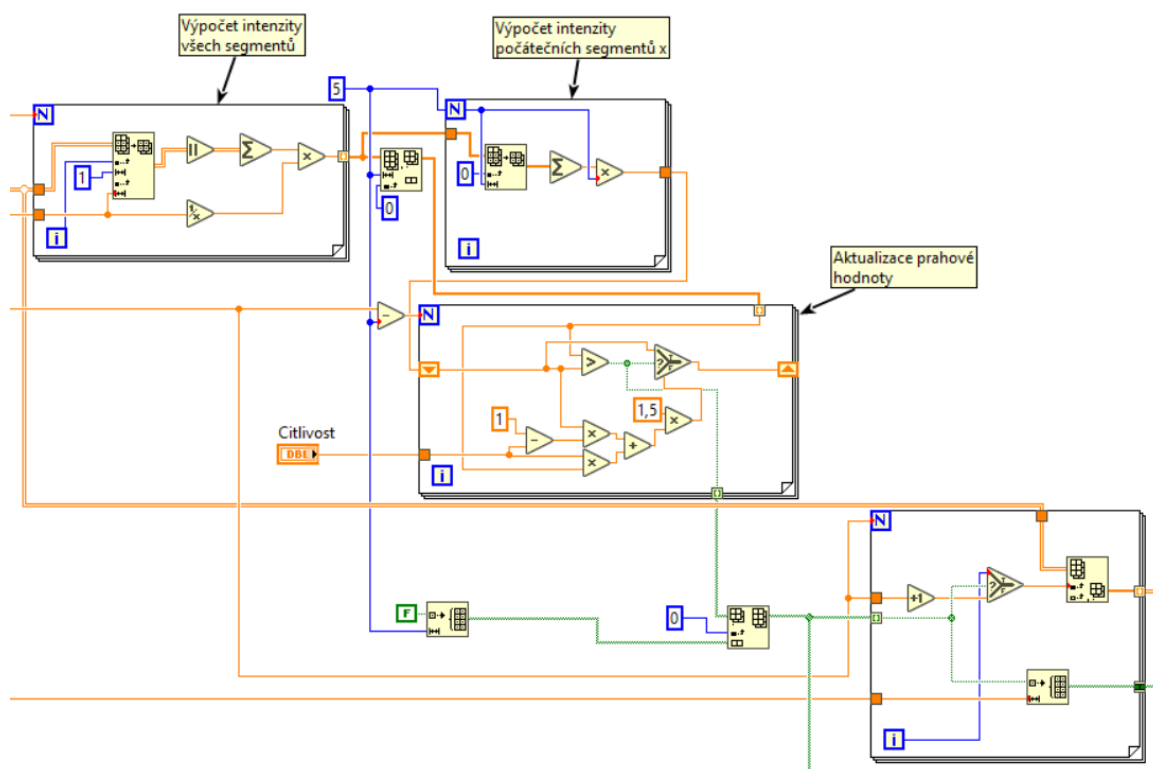
Poslední částí vývojového diagramu představuje aktualizaci prahové hodnoty podle vzorce

$$M_n = (1 - p)M_d + pM. \quad (1.46.)$$

Hodnota  $p$  vyjadřuje citlivost detekce, která se nejčastěji volí v rozmezí od 0,4 do 0,8. Aktualizace je prováděna v případě, jeli hodnota intenzity právě testovaného segmentu menší, než je hodnota prahová. V tomto případě se jedná o řečově neaktivní segment a do proměnné  $DET\_SEG$  je zapsána hodnota 0. Tento vektor slouží jako registr segmentů. V případě detekce řečové aktivity se aktualizace provádět nebude a hodnota bude přenesena do dalšího cyklu. Do  $DET\_SEG$  je zapsána hodnota 1. Hodnoty jsou ukládány jako sloupcové vektory, které slouží k dalšímu zpracování.

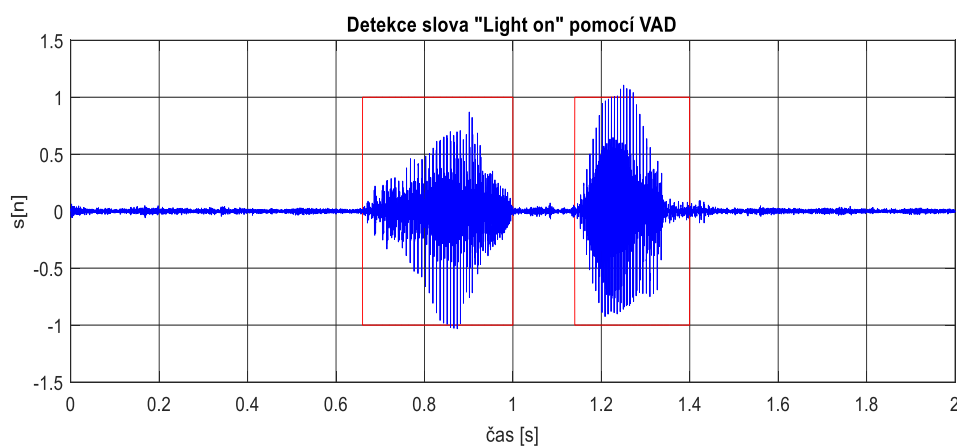


Obr. 23: Vývojový diagram inicializace a průběhu detekce intenzitního detektoru.



Obr. 24: Implementace VAD v LabVIEW.

Obr. 24 zobrazuje implementaci detektoru řečové aktivity v prostředí LabVIEW. Je zde zobrazen výpočet intenzity jednotlivých segmentů, počáteční prahové hodnoty a aktualizace prahové hodnoty. Dále se zde počítá celkový počet segmentů ( $DET\_SEG$ ), ve kterých byla detekována řeč a jejich přesné pořadí, aby bylo možné tuto detekci znázornit do grafu. Po detekci řeči, jejíž příklad lze pozorovat v grafu (Obr. 25), jsou vypočtené hodnoty přesunuty do další fáze a je vypočtena hodnota  $SSNR$ .



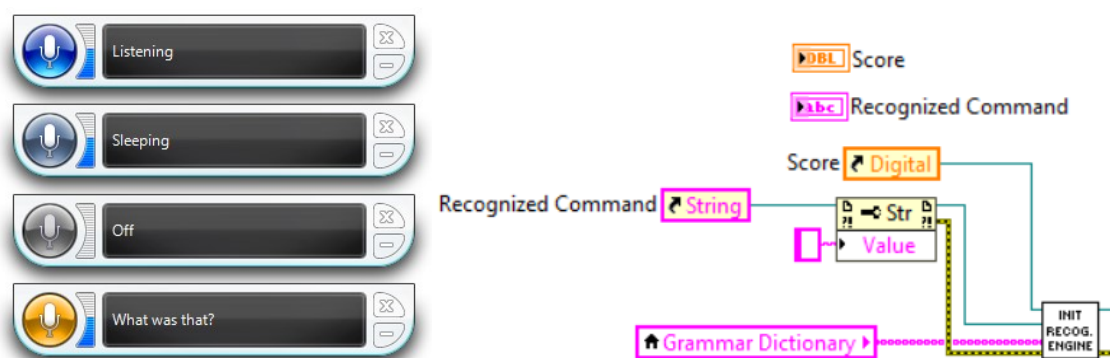
Obr. 25: Detekce slova pomocí VAD.



#### 4.4.2 Realizace rozpoznávače řeči

Jedním z cílů diplomové práce bylo hodnotit výsledky filtrace jak pomocí hodnot odstupe signálu od šumu  $SNR$ , tak i za použití rozpoznávače řeči WSR. Rozpoznávač řeči použitý v diplomové byl od firmy Microsoft a jedná se o Windows Speech Recognizer (WSR) [58]. Rozpoznávač řeči je komerčně dostupný a volně stažitelný ze stránek výrobce [59]. Rozpoznávač WSR se tedy dá stáhnout jako aplikace do každého počítače, jehož operační systém je novější jak Windows Vista. Vzhled této aplikace je zobrazen na obrázku vlevo (Obr. 26).

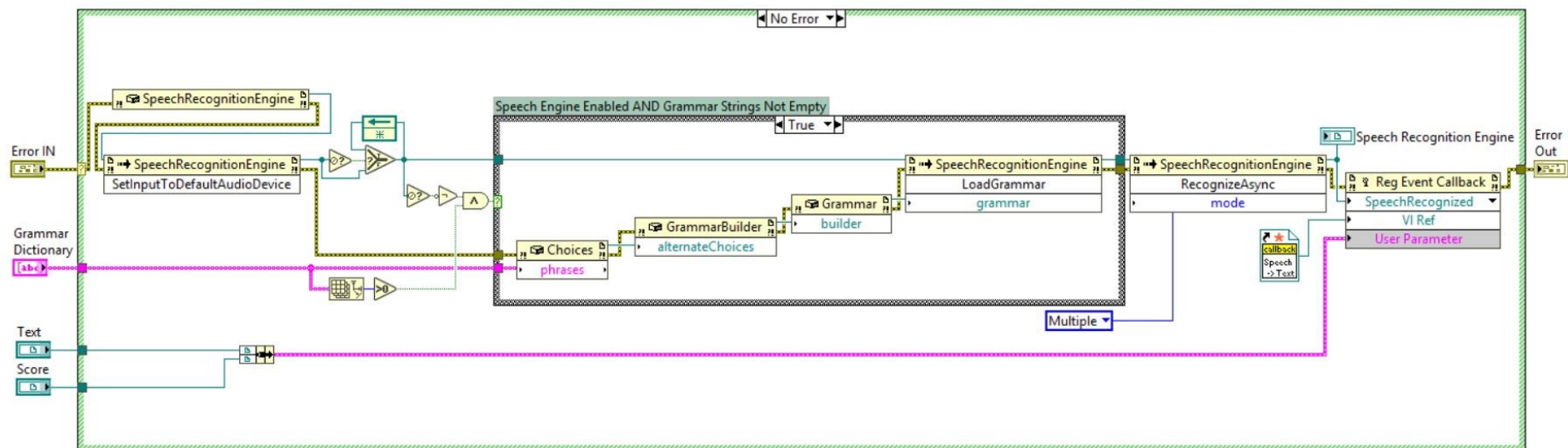
Rozpoznávač řeči je lokálně zpracovávaná platforma, která se přizpůsobuje na základě uživatelského kontextu, gramatiky a ukázek řeči. Přesnost rozpoznávače se zvyšuje jeho použitím, což pomáhá přizpůsobit se gramatice uživatele, jeho slovní zásobě a použití slov. Přesnost rozpoznávače se také zvýšit pomocí výukového programu.



Obr. 26: Aplikace rozpoznávače řeči (vlevo) a jeho implementace v Labview.

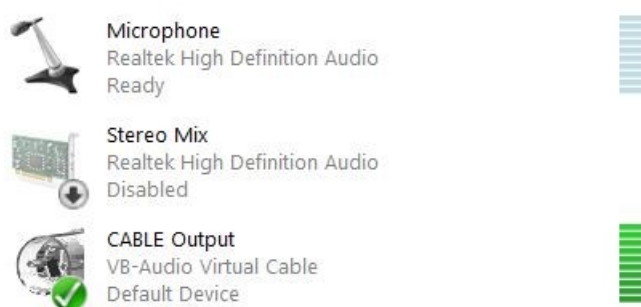
Pomocí této aplikace může uživatel ovládat svůj počítač jen za pomoci hlasových příkazů nebo například diktovat text do dokumentů a e-mailů. Jelikož se jedná o zahraniční firmu, preferuje tento rozpoznávač jazyky jako je angličtina, čínština, francouzština, němčina a další. Pro jazykovou sadu češtiny má pouze omezenou slovní zásobu a nedokáže jednotlivé příkazy tak snadno odlišit. Musí se tedy pro počítač, ve kterém bude aplikace používána nastavit konkrétní jazyk. V mém případě se jednalo o jazyk angličtina, a proto jsou příkazy, na kterých budou prováděny výsledky diplomové práce nahrány v tomto jazyce.

Jelikož bylo nutné pro výslednou aplikaci pracovat pouze s vnitřní částí tohoto programu, a ne s celou aplikací, musel být stažen spolu s aplikací i engine, jehož implementaci ve vývojovém prostředí LabVIEW, lze vidět na předchozím obrázku vpravo. Tento engine (Init Recognition Engine) již není volně stažitelný ze stránek výrobce, ale musel být vyhledán na fórech. Obr. 27 zobrazuje, jak vypadá jeho vnitřní část. Tím že se jedná o komerčně dostupný rozpoznávač, je to zároveň i jeho nevýhoda. Tento rozpoznávač se nedá programově upravovat. Firma Microsoft si totiž chrání veškerou vnitřní strukturu, takže se nedá zjistit na jakém principu pracuje. Uživatel si pouze na startu zvolí, jaké slova mají být rozpoznávány pomocí proměnné „Grammar dictionary“ a poté na výstupu „Recognized command“ uvidí jaké slovo, nebo fráze byla právě rozpoznána.



Obr. 27 Engine rozpoznávače řeči WSR

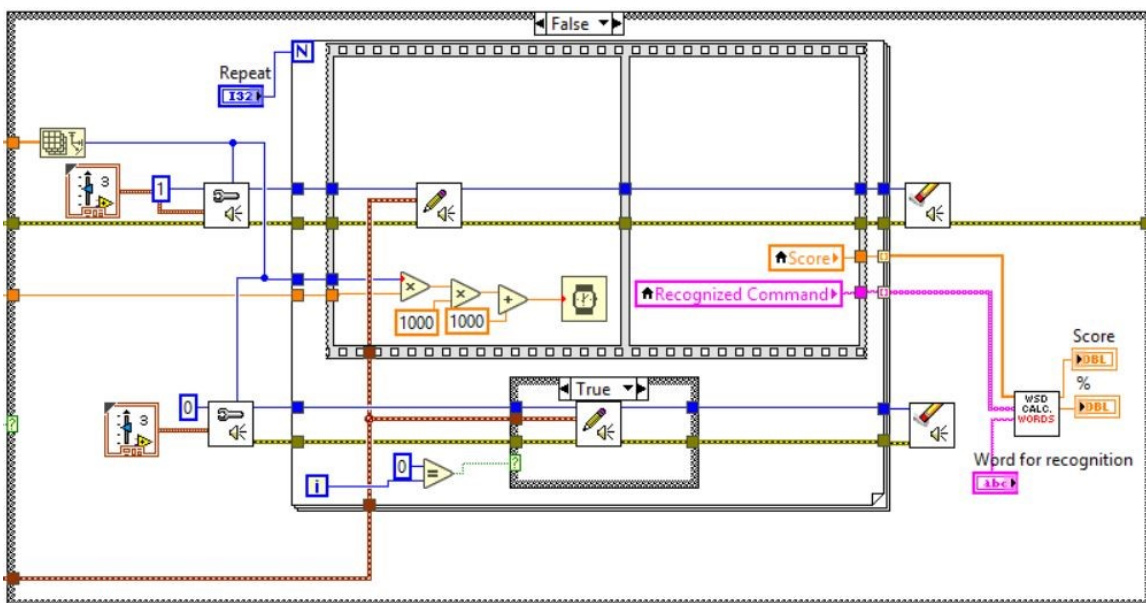
Jelikož bude rozpoznávač pracovat na principu již předem nahraných frází, a ne přímo se vstupem konkrétního mikrofону, musela být použita ještě jedna aplikace. Jedná se o aplikaci VB-CABLE [60], pomocí níž se v počítači vytvoří zařízení, které pracuje jako virtuální zvukový kabel. Na Obr. 28 níže je vidět, jak toto zařízení vypadá v nastavení zvuku v počítači. Je nutné, aby zařízení bylo nastaveno jako výchozí před startem aplikace.



Obr. 28: VB-CABLE Virtual Audio Device.

Veškeré použité programy, které byly zmíněny, se nacházejí v dokumentu v sekci přílohy a musejí být staženy a nainstalovány před spuštěním výsledné aplikace do provozu. Dále musí být nastaven jazyk angličtina pro zvolený počítač, na kterém bude aplikace provozována.

Na Obr. 29 je zobrazena implementace virtuálního kabelu. V části, která slouží pro zápis dat do virtuálního kabelu je vytvořeno zpoždění. Kdyby se v programu nevyskytovalo zpoždění nahrávky by byly zpracovávány příliš rychle a rozpoznávač by nedokázal vyhodnocovat výsledky. V pravé části se nachází SubVI *WSD\_Calc.\_Words*, které je v programu vytvořeno pro vyhodnocení rozpoznávaných slov. Toto vyhodnocení je porovnáno se slovem, které uživatel zapsal do aplikace v jeho druhé části *Word for recognition*. V programu je poté zobrazena úspěšnost detekce v procentech a průměrné skóre rozpoznávače, kterého při vyhodnocení dosáhnul. Ve spodní části implementace je vytvořen i výstup do reproduktorů. Uživatel si tak může přehrát čistou řečovou nahrávku, nahrávku kontaminovanou hlukem a výsledek filtrace. Poté může uživatel porovnat dané nahrávky a zjistit tak úspěšnost filtrace z pohledu subjektivního hodnocení řeči nebo podle objektivního hodnocení řeči, pomocí hodnot *SNR*. Při zvolení vyššího počtu opakování, se nahrávka přehraje pouze jednou.



Obr. 29: Implementace virtuálního kabelu.

## 4.5 Testování rozpoznávače WSR

Jak již bylo zmíněno dříve, nedá se přesně zjistit, jak takovýto rozpoznávač pracuje v jeho vnitřní struktuře. Jedná se tedy o „black box“, kterému poskytneme vstupní data a on nám vrátí slovo nebo slovní spojení, které rozpoznal. Byla proto uskutečněna série testů, aby se zjistilo, s jakými signály na vstupu dokáže tento rozpoznávač pracovat. Pro testy byly použity nahrávky do domácnosti a celkový počet opakování pro každou nahrávku byl 50.

První test v Tab. 2 byl spojený s úspěšností rozpoznávání před filtrací a po filtraci. V obou případech se jednalo o nahrávky čisté řeči, které byly pořízeny v bezdozvukové komoře. Pod pojmem čistá řeč, se jedná o nahrávky řeči, které byly upraveny pouze co se týká amplitudy na hodnoty v rozmezí od -1 do 1. Druhý sloupec zobrazuje hodnoty čisté řeči, na kterých byla provedena nejen úprava amplitudy, ale byla zde i použita filtrace pomocí filtru Butterworth, který byl nastaven na 5. řád, pásmovou propust v rozmezí od 280 Hz do 3500 Hz, jelikož se ve vedlejších pásmech již nevyskytují užitečné informace o řeči.

Z tabulky lze pozorovat, že úspěšnost rozpoznávání slov nebyla vůbec dobrá. Jednalo se totiž o čisté nahrávky řeči, které nebyly žádným způsobem zkresleny, i tak byla průměrná hodnota rozpoznání 82 %. Po zjištění ideálního nastavení filtru, tak aby rozpoznávač pracoval co nejlépe, lze pozorovat, že se rozpoznávání jednotlivých frází výrazně zlepšilo a průměrná úspěšnost rozpoznávání již byla 98 %.

Tab. 2: Úspěšnost rozpoznávače řeči, pomocí filtrace.

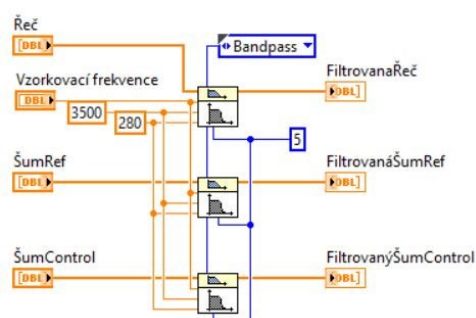
Nahrávka	Čistá řeč [%]	Po filtraci [%]
Light on	100	100
Light off	45	95
Blinds up	78	98
Blinds down	100	100
Volume up	65	95
Volume down	100	100
Increase temperature	71	98
Decrease temperature	99	100

Druhý test, který byl proveden na rozpoznávač bylo testování vstupního rozsahu signálu. Výsledky testu jsou zobrazeny v Tab. 3. Celkový počet opakování při jednotlivých hodnotách amplitudy byl 50. Z výsledků lze pozorovat, že pokud poskytneme rozpoznávači menší rozsah hodnot amplitud, než je  $\langle -0,1; 0,1 \rangle$ , rozpoznávač přestane pracovat. To stejné platí i pro příliš vysoké hodnoty amplitud nad rozsah  $\langle -10; 10 \rangle$ .

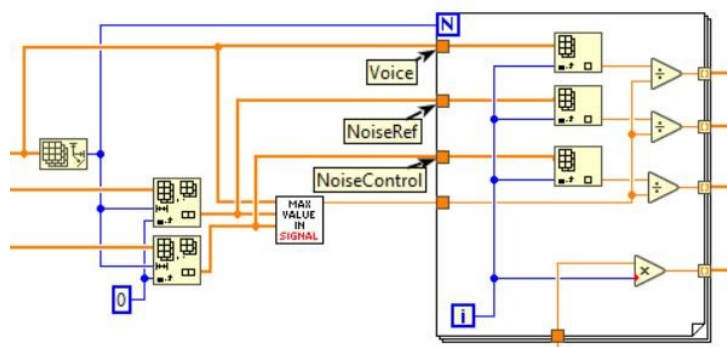
Tab. 3: Úspěšnost rozpoznávání řeči na příkazu „Light on“ při změně amplitudy.

Amplituda [-]	Rozpoznání [%]
$\langle -0,01; 0,01 \rangle$	0
$\langle -0,05; 0,05 \rangle$	0
$\langle -0,1; 0,1 \rangle$	100
$\langle -0,5; 0,5 \rangle$	100
$\langle -1; 1 \rangle$	100
$\langle -5; 5 \rangle$	100
$\langle -10; 10 \rangle$	100
$\langle -50; 50 \rangle$	0
$\langle -100; 100 \rangle$	0

Z předchozích zjištění tedy vyplývá, že rozpoznávač nepracuje vždy, a se všemi hodnotami stejně. Má svůj určitý rozsah hodnot, ve kterém pracuje. Proto budou všechny nahrávky upravovány podle předem zjištěných faktů. Bude použita předfiltrace pomocí Butterworth filtru (Obr. 30) a signál bude normalizován na hodnotu v rozsahu  $\langle -1; 1 \rangle$  (Obr. 31).



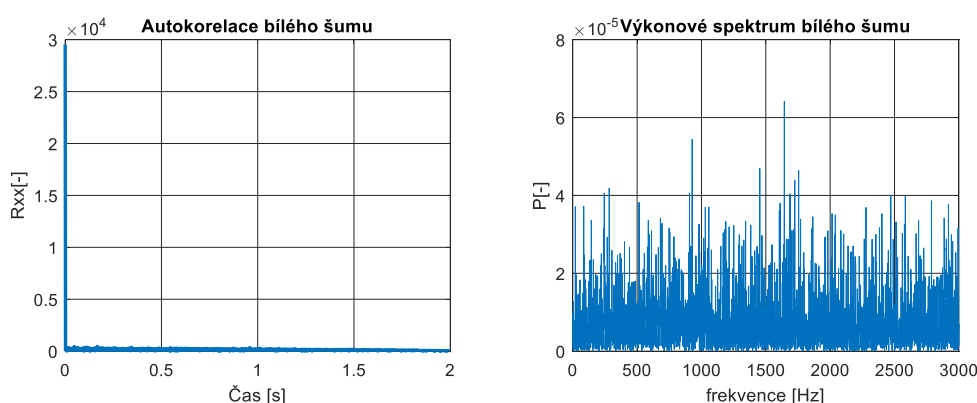
Obr. 30: Implementance Butterworth filtru.



Obr. 31: Implementace úpravy amplitudy.

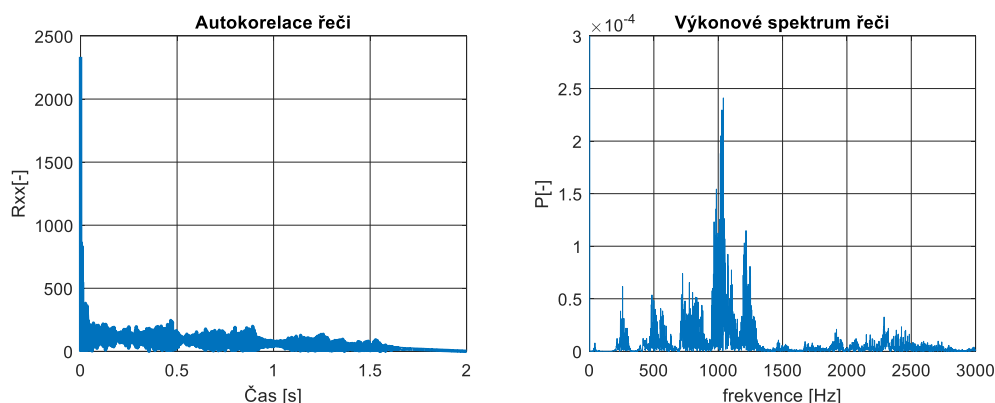
## 4.6 Rozbor použitých šumů

Na následujících obrázcích lze pozorovat porovnání autokorelačních funkcí a výkonových spekter pro nahrávky typu, bílý šum, řeč, šum myčky a benzínového automobilu. První graf (Obr. 32) zobrazuje autokorelaci a výkonové spektrum bílého šumu, kde lze pozorovat, že hodnoty autokorelace jsou nulové a vyskytuje se zde jedna hodnota  $R[0]$ , která je dána rozptylem šumu. Na grafu vpravo je zobrazeno výkonové spektrum, kde lze pozorovat, že má téměř konstantní hodnoty, které jsou zastoupeny v celém frekvenčním pásmu. Zejména proto je tento druh šumu velice obtížné filtrovat.



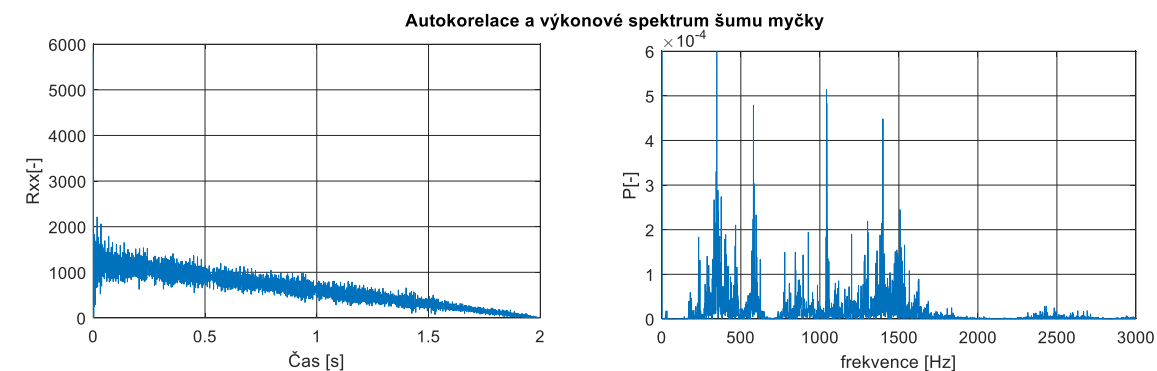
Obr. 32: Bílý šum.

Druhý graf (Obr. 33) zobrazuje opět autokorelaci a výkonové spektrum pro nahrávku řeči. Autokorelační funkce již nemá nenulové hodnoty, jak tomu bylo u bílého šumu. Ve výkonovém spektru lze pozorovat výrazné změny oproti bílému šumu. Nahrávka řeči „Turn on the press machine“ se vyskytuje pouze v rozmezí od 250 Hz do 3000 Hz. Tedy v pásmu, kde se vyskytují nejdůležitější informace v řeči.

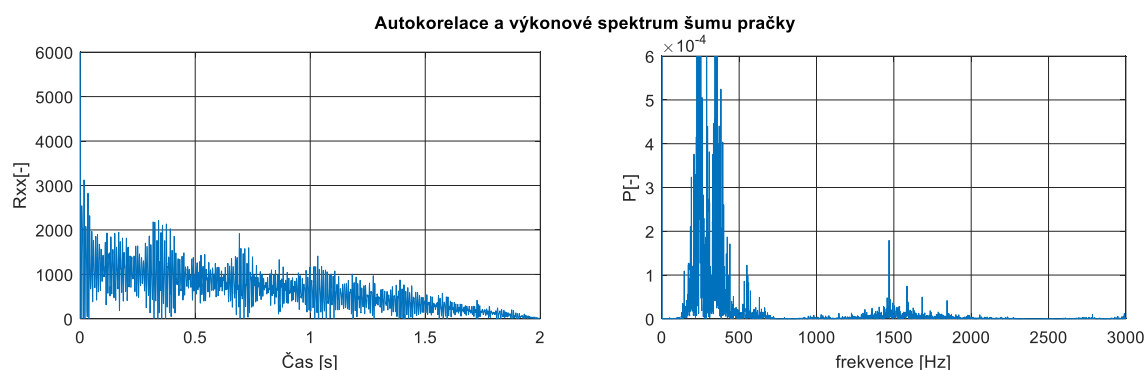


Obr. 33: Řeč "Turn on the press machine".

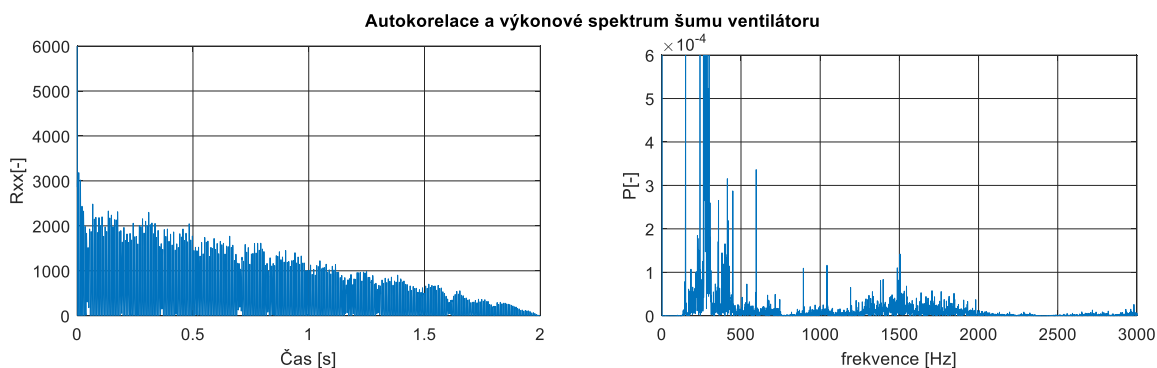
V následujících grafech (Obr. 34) jsou zobrazeny autokorelace a výkonová spektra pro tři druhy šumů. Tyto šумы se v diplomové práci používají pro hodnocení úspěšnosti filtrace v domácnosti. Jedná se o šum myčky, pračky a ventilátoru. Z výkonových spekter je patrné, v jakém frekvenčním pásmu se daný šum nachází a jaký je mezi nimi frekvenční rozdíl. Šum pračky se pohybuje spíše v nižších frekvencích, přibližně do hodnoty 400 Hz. Zatímco šum myčky nebo ventilátoru se již vyskytuje ve frekvenčním pásmu řeči. Z těchto údajů vyplývá, že filtrace šumu myčky nebo ventilátoru bude dosahovat horších výsledků než u šumu pračky.



a)



b)



c)

Obr. 34: Autokorelace a výkonové spektrum pro šum a) myčky, b) pračky, c) ventilátoru.



## 4.7 Výsledky diplomové práce

V následující kapitole budou popsány výsledky pro filtraci pomocí adaptivních filtrů. Na Obr. 35 je zobrazena implementace jednoho ze čtyř algoritmů (LMS, NLMS, RLS, QR-RLS), které má uživatel k dispozici a které jsou v programu zpracované.

Postupně bude za pomocí algoritmů LMS a RLS zpracována úspěšnost filtrace při rozpoznávání hlasových příkazů v domácnosti, v automobilu, továrně a v poslední řadě i letadle. Úspěšnost filtrace bude posuzována pomocí hodnot  $SNR$  a  $SSNR$ . Dále pak úspěšnost filtrace posuzována i za pomocí rozpoznávače řeči WSR. V tabulkách bude ukázáno, jak si rozpoznávač dokázal s těmito poradit před filtrací a po filtraci.

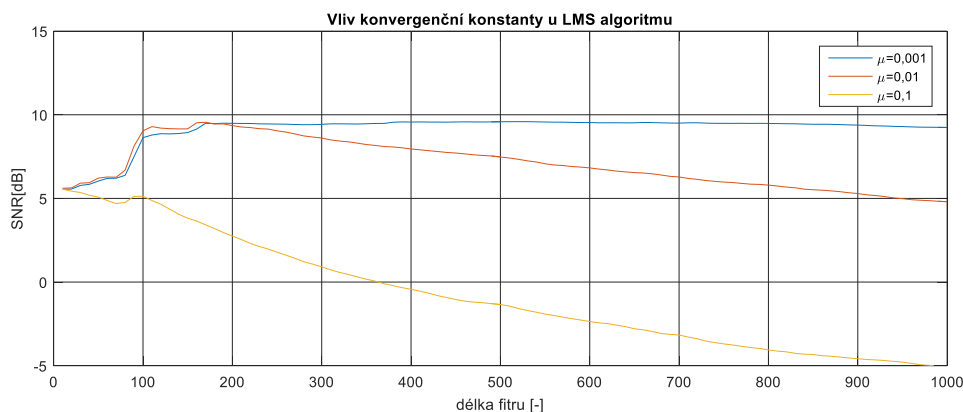


### 4.7.1 Použití v domácnosti

Pro posouzení kvality filtrace bylo vybráno 8 příkazů se kterými může uživatel přijít do styku při hlasovém ovládání domácnosti. Zde jsou uvedeny jednotlivé příkazy:

- Light on.
- Light off.
- Blinds up.
- Blinds down.
- Volume up.
- Volume down.
- Increase temperature.
- Decrease temperature.

Vytvořený program pracuje automatizovaně co se týká výběru neoptimálnějších parametrů pro nastavení daného algoritmu. Na Obr. 36 je zobrazen právě vliv konvergenční konstanty  $\mu$  na délce filtru pro použitý algoritmus LMS. Konvergenční konstanta představuje malou kladnou hodnotu, která ovlivňuje vlastnosti adaptace. V grafu lze pozorovat, je-li tato konstanta příliš velká, dochází k rozvažování vah. Konvergenční konstanta musí být nastavena co neoptimálněji ke vstupnímu signálu a souvisí tedy s velikostí jeho rozsahu. V grafu lze pozorovat rychlý nárůst hodnot  $SNR$  pro velikosti konvergenčních konstant 0,001 a 0,01, zatímco nejvyšší hodnota konvergenční konstanty 0,1 se dostala při zvyšování délky filtru až do záporných hodnot  $SNR$ .



Obr. 36: Vliv konvergenční konstanty  $\mu$ .

V následujících tabulkách (Tab. 4, Tab. 5, Tab. 6) jsou uvedeny výsledky pro filtraci pomocí LMS algoritmu. Tabulky jsou rozděleny na tři části. V první části jsou zobrazeny hodnoty řádu filtru. Tyto hodnoty jsou uvedeny jako ideální z pohledu nejvyšší hodnoty  $SNR$  po filtraci. Druhá část zobrazuje hodnoty  $SNR$ ,  $SSNR$  a úspěšnost rozpoznávače řeči před provedenou filtrací. V poslední části jsou zobrazeny hodnoty  $SNR$ ,  $SSNR$  a úspěšnost WSR po provedené filtraci pomocí LMS algoritmu.

Tab. 4: Výsledky filtrace a úspěšnost rozpoznávače pro šum pračky (LMS).

Filtrace LMS	Pračka maximální hlasitost, $\mu = 0,01$						
	Řád filtru M [-]	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Light on	110	3,28	10,44	0	15,72	21,24	95
Light off	170	-1,25	5,99	0	13,50	17,63	96
Blinds up	170	0,27	8,24	0	14,02	22,49	98
Blinds down	110	3,21	9,28	2	16,21	22,63	97
Volume up	170	3,43	8,73	5	16,62	22,54	82
Volume down	190	5,10	10,64	85	18,15	24,13	99
Increase temperature	170	-2,18	2,52	0	11,80	15,70	25
Decrease temperature	170	-4,20	0,67	0	10,70	13,98	90

Tab. 5: Výsledky filtrace a úspěšnost rozpoznávače pro šum ventilátoru (LMS).

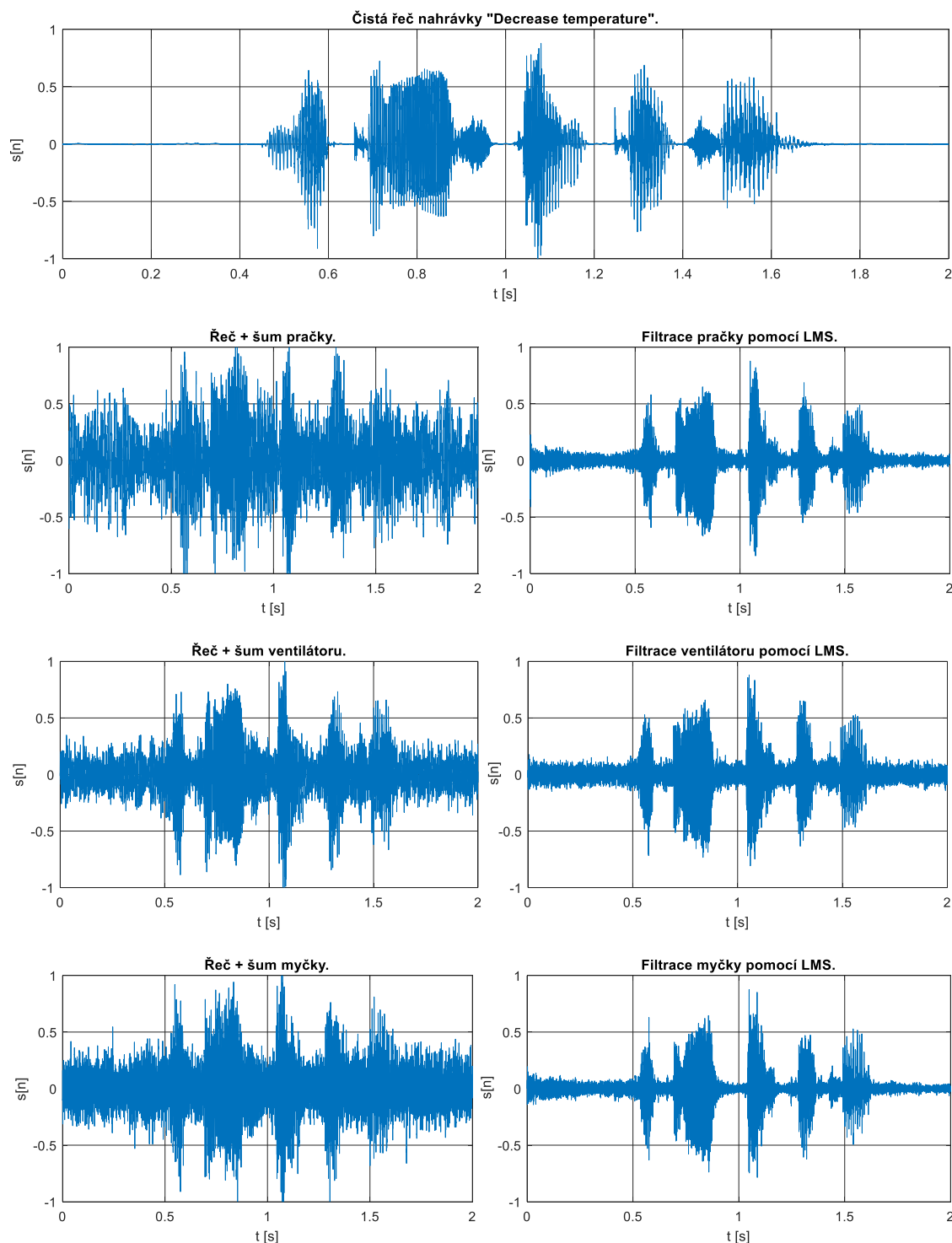
Filtrace LMS	Ventilátor maximální hlasitost, $\mu = 0,01$						
	Řád filtru M [-]	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Light on	160	10,89	17,53	0	16,59	20,35	89
Light off	110	7,61	12,58	0	12,35	15,99	28
Blinds up	170	9,14	14,49	10	13,42	18,04	25
Blinds down	170	12,07	15,43	1	16,33	19,55	82
Volume up	170	11,29	15,11	7	14,71	18,07	79
Volume down	110	13,96	15,12	89	17,83	21,53	72
Increase temperature	170	6,68	11,02	0	11,08	14,48	15
Decrease temperature	140	4,66	7,88	1	9,31	11,06	97

Tab. 6: Výsledky filtrace a úspěšnost rozpoznávače pro šum myčky (LMS).

Filtrace LMS	Myčka maximální hlasitost, $\mu = 0,01$						
	Řád filtru M [-]	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Light on	30	10,51	15,91	35	17,00	21,82	89
Light off	30	5,98	10,63	55	12,79	16,99	25
Blinds up	30	7,51	12,40	0	13,80	18,92	5
Blinds down	30	10,44	13,80	0	16,37	20,32	28
Volume up	30	10,67	13,45	2	16,33	18,74	34
Volume down	30	12,33	15,94	85	18,14	19,77	65
Increase temperature	40	5,05	9,00	1	12,00	15,45	15
Decrease temperature	30	3,03	5,63	0	10,19	12,10	99

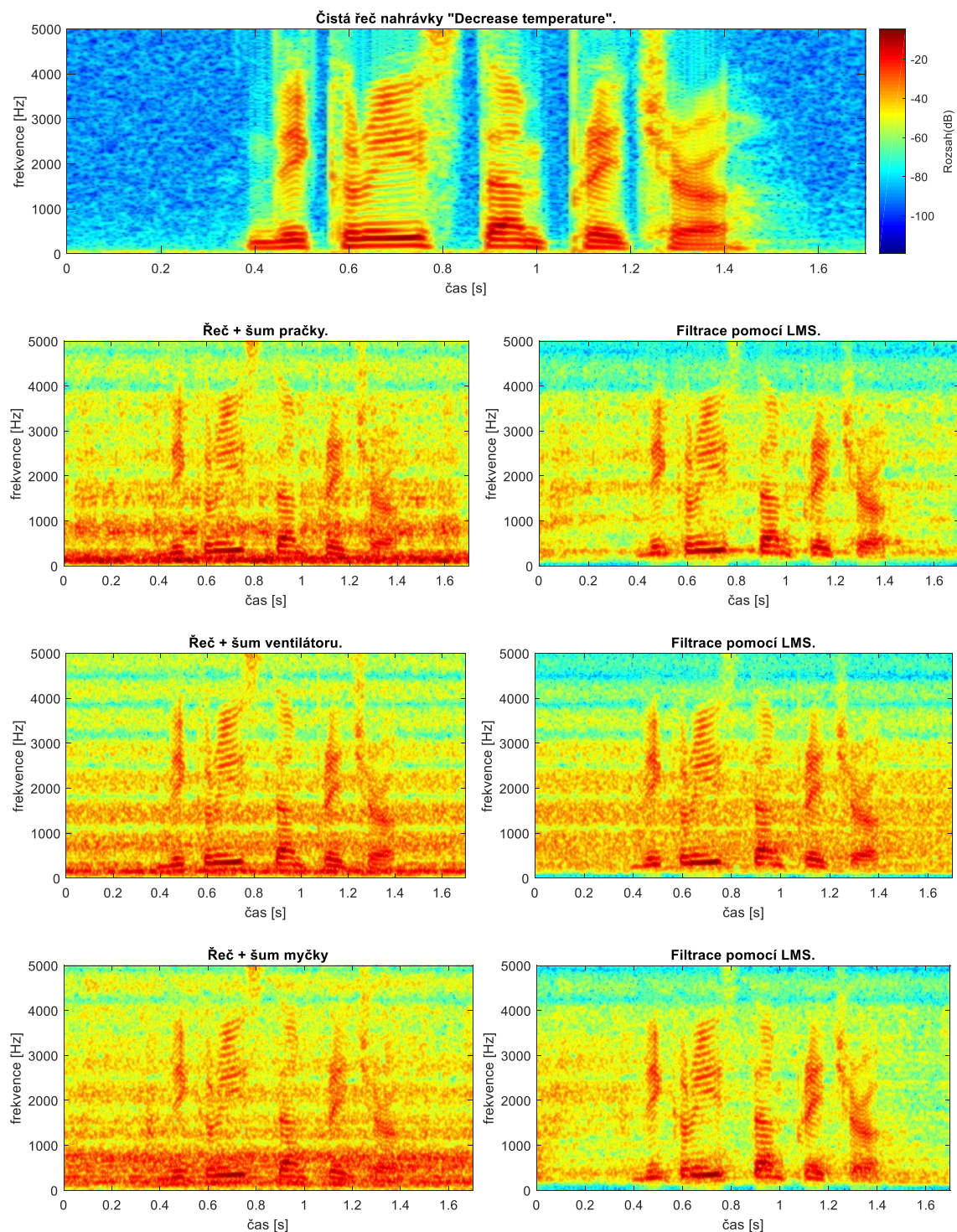
Úspěšnost filtrace byla testována na třech různých hlucích, se kterými můžeme běžně přijít do styku v domácnosti. Konkrétně byl vybrány šum pračky, ventilátoru a myčky, při jejich nejvyšší intenzitě hlasitosti. Pro každou testovanou nahrávku byla úspěšnost rozpoznání testována stokrát. Z tabulek lze pozorovat úspěšnost rozpoznávače z hlediska *SNR* a rozpoznání WSR. Nejlépe si rozpoznávač vedl s odfiltrováním šumu pračky, kde byla výsledná úspěšnost téměř vždy vyšší než 90 %. Pro šum myčky a ventilátoru nebyl rozpoznávač ani v jednom z případů příliš úspěšný. Úspěšnost se totiž pohybovala v průměru kolem 60 % pro ventilátor a 45 % pro šum myčky. Zatímco u pračky byla úspěšnost v průměru 85 %.

Dalším výsledkem předchozích tabulek je úspěšnost filtrace z pohledu hodnot *SNR*. Průměrné zlepšení hodnoty *SNR* pro šum pračky bylo 13,6 dB, pro šum ventilátoru 4,4 dB a myčku 6,4 dB. Pro hodnotu *SSNR*, tedy jen pro řečově aktivní úseky bylo vidět zlepšení pro pračku o 12,98 dB, pro ventilátor o 3,74 dB a pro šum myčky to bylo 5,92 dB. Podle těchto výsledků byl šum, který byl nejlépe odfiltrován, právě šum pračky, pro který rozpoznávač pracoval nejlépe.



Obr. 37: Časové průběhy před a po filtraci za použití LMS algoritmu.

Obr. 37 zobrazuje výsledek filtrace LMS algoritmu. Je zde zobrazena nahrávka „Decrease temperature“, která je kontaminována všemi druhy hluků, které byly použity. V levé části se nachází nahrávky kontaminované hlukem a v pravé části jsou zobrazeny časové průběhy po filtraci za použití LMS algoritmu. Obr. 38 zobrazuje spektrogramy daných nahrávek. Lze pozorovat potlačení šumu, které je z grafů nejvýraznější u šumu pračky a myčky.



Obr. 38: Testované signály v časově-frekvenční oblasti před a po filtraci pomocí LMS algoritmu znázorněné ve spektrogramu.

Následně byla provedena filtrace za použití druhého algoritmu, a to RLS. U tohoto algoritmu byla stanovena hodnota faktoru zapomínání  $\lambda = 1$ , což představuje nekonečnou paměť. Dále zde byl jednotně nastaven řád filtru na hodnotu 200. Algoritmus RLS se vyznačuje vysokou matematickou náročností oproti LMS, a to představuje potíže, při použití tohoto algoritmu v reálném prostředí. Tato matematická náročnost jde vidět na Obr. 42.

Tab. 7: Výsledky filtrace a úspěšnost rozpoznávače pro šum pračky (RLS).

Filtrace RLS	Pračka maximální hlasitost, $\lambda = 1$ , $M = 200$					
	Před filtrací			Po filtraci		
	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Light on	3,28	10,44	0	19,23	22,63	99
Light off	-1,25	5,99	0	16,40	21,27	97
Blinds up	0,27	8,24	0	16,34	21,83	94
Blinds down	3,21	9,28	2	19,35	23,46	99
Volume up	3,43	8,73	5	19,22	22,94	97
Volume down	5,10	10,64	85	21,17	25,04	96
Increase temperature	-2,18	2,52	0	14,83	18,78	94
Decrease temperature	-4,20	0,67	0	13,18	17,32	100

Tab. 8: Výsledky filtrace a úspěšnost rozpoznávače pro šum ventilátoru (RLS).

Filtrace RLS	Ventilátor maximální hlasitost, $\lambda = 1$ , $M = 200$					
	Před filtrací			Po filtraci		
	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Light on	10,89	17,53	0	18,21	21,77	100
Light off	7,61	12,58	0	13,76	17,59	13
Blinds up	9,14	14,49	10	15,13	19,87	100
Blinds down	12,07	15,43	1	17,77	20,83	89
Volume up	11,29	15,11	7	17,46	19,97	98
Volume down	13,96	15,12	89	17,56	20,26	100
Increase temperature	6,68	11,02	0	12,47	16,02	22
Decrease temperature	4,66	7,88	1	10,64	12,90	89



Tab. 9: Výsledky filtrace a úspěšnost rozpoznávače pro šum myčky (RLS).

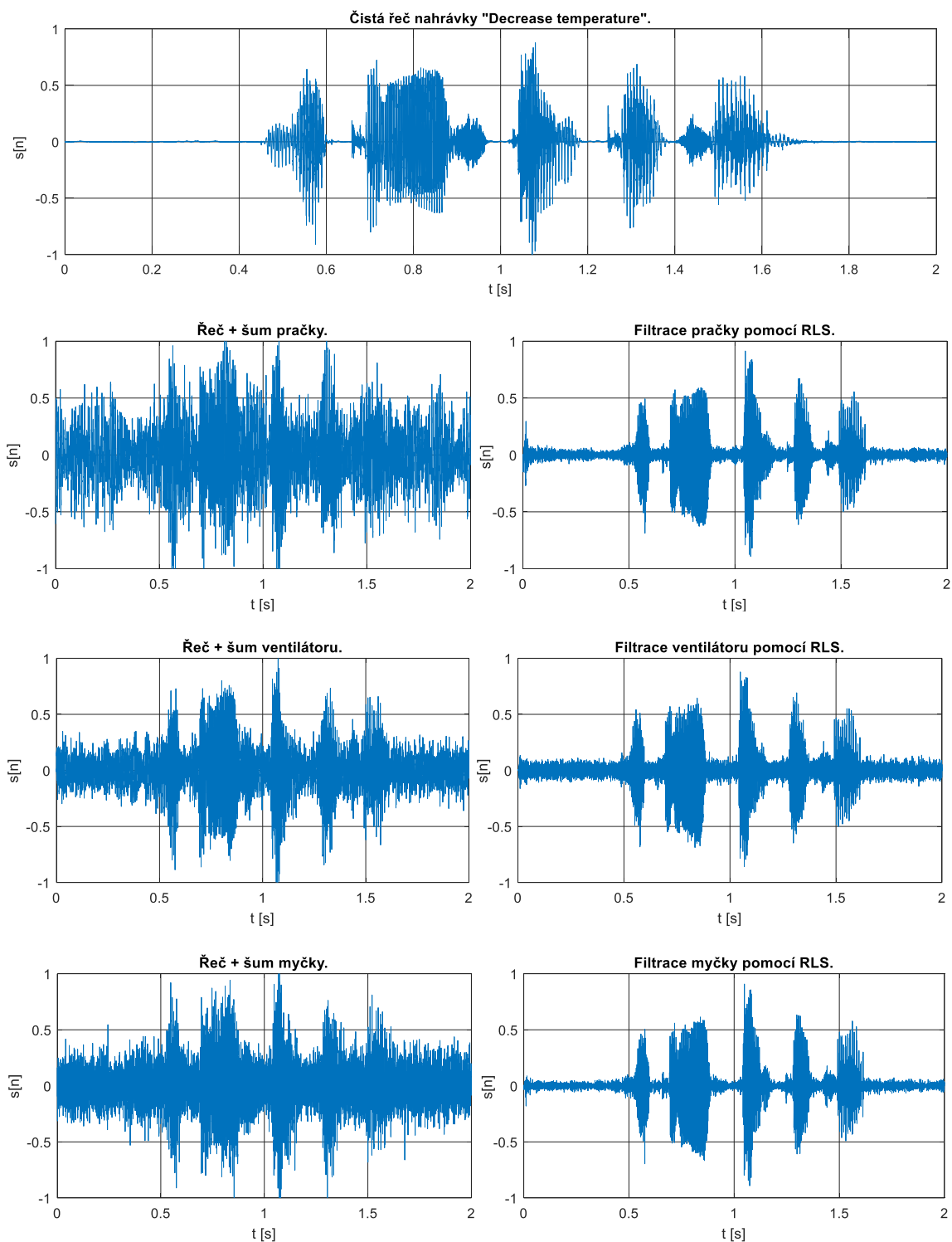
Filtrace RLS	Myčka maximální hlasitost, $\lambda = 1$ , $M = 200$					
	Před filtrací			Po filtraci		
	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Light on	10,51	15,91	35	20,42	23,98	100
Light off	5,98	10,63	55	17,25	21,54	96
Blinds up	7,51	12,40	0	17,58	22,16	100
Blinds down	10,44	13,80	0	20,63	23,47	98
Volume up	10,67	13,45	2	20,67	23,24	99
Volume down	12,33	15,94	85	21,2	24,11	100
Increase temperature	5,05	9,00	1	16,33	20,15	95
Decrease temperature	3,03	5,63	0	14,15	16,88	100

V Tab. 7, Tab. 8, Tab. 9 se nacházejí výsledky pro filtrace pomocí RLS algoritmu. Stejně jako u LMS jsou zde zobrazeny hodnoty před filtrací, po filtraci. Tabulky jsou tedy rozděleny na dvě části. Filtrace byla testována pro celkem tři druhy šumů. Každá nahrávka byla opakována celkem stokrát. Po filtraci pracoval rozpoznávač nejlépe pro šum pračky a myčky. Kdy výsledky byly téměř stejné a to 97 % pro šum pračky a 98 % pro šum myčky. Pro šum ventilátoru již nebyl rozpoznávač příliš úspěšný. Jeho úspěšnost byla 76 %. Výsledná úspěšnost je nízká z toho důvodu, že rozpoznávač nedokázal správně vyhodnotit dva druhy nahrávek.

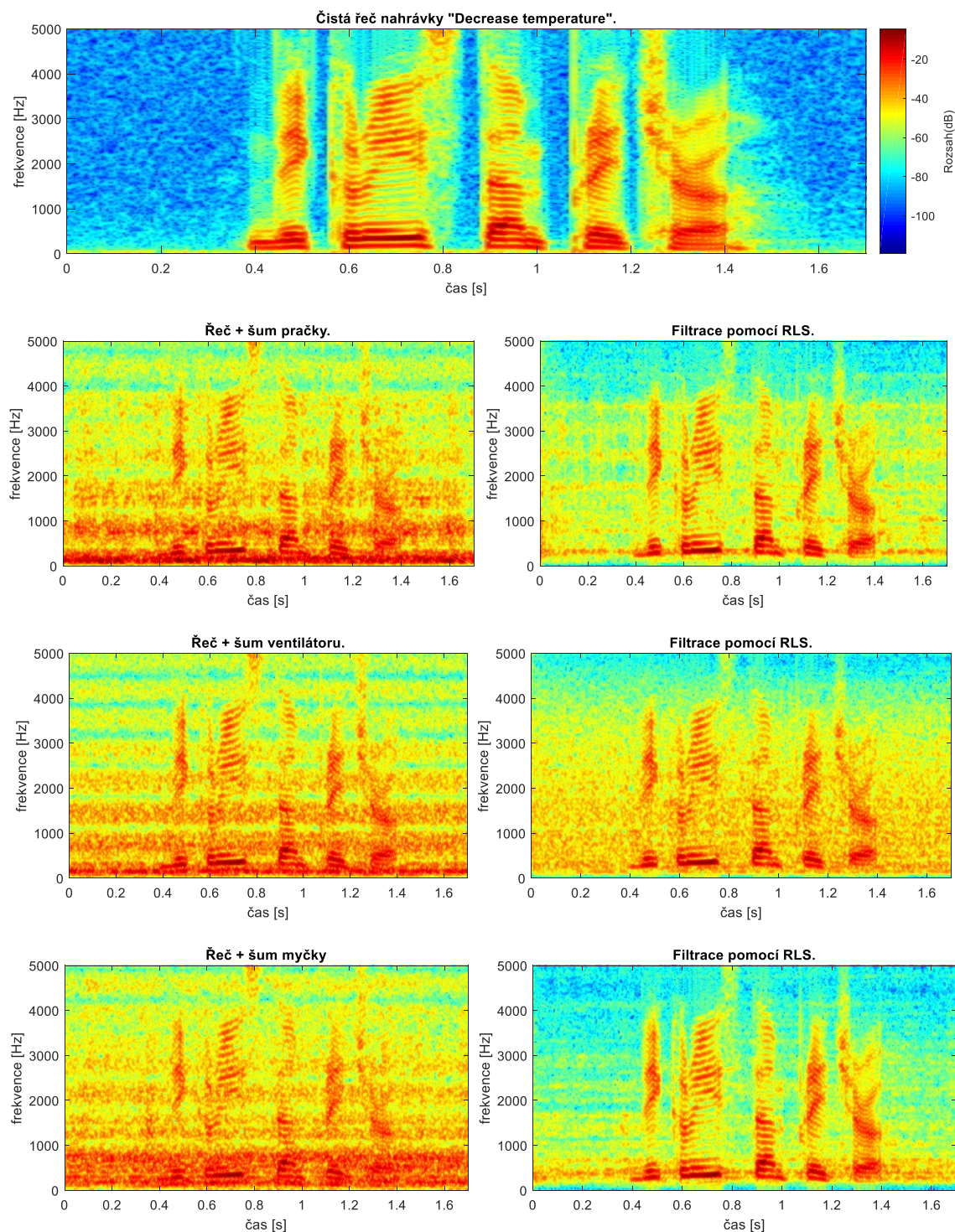
Dalším výsledkem předchozích tabulek je úspěšnost filtrace pomocí hodnot  $SNR$  a  $SSNR$ . Pro šum pračky bylo průměrné zlepšení hodnoty  $SNR$  o 16,51 dB, pro šum ventilátoru 5,84 dB a u myčky lze vidět zlepšení o 10,34 dB. Co se týká hodnot  $SSNR$  je výsledné zlepšení pro šum pračky 14,6 dB, pro šum ventilátoru 5 dB a pro myčku 9,85 dB. Podle výsledků  $SNR$  a  $SSNR$ , lze pozorovat u šumů pračky a myčky nejvyšší zlepšení. Právě pro tyto druhy šumů byla i nejvyšší úspěšnost rozpoznávače.

V grafech (Obr. 39, Obr. 40) jsou zobrazeny výsledky filtrace. První obrázek zobrazuje graf pro nahrávku „Decrease temperature“. V levé části je zobrazena daná nahrávka, která je smíchána všemi druhy šumů. Následně je v pravé části zobrazena výsledná filtrace za použití RLS algoritmu.

Druhý obrázek zobrazuje výsledek filtrace pro stejnou nahrávku v časově-frekvenční doméně. Pomocí těchto výsledků, můžeme pozorovat odstup signálu od šumu v grafické podobě. Nejlépe si RLS algoritmus vedl právě se šumy pračky a myčky, přesně tak jak ukazují výsledky  $SNR$  a úspěšnosti rozpoznávače WSR.



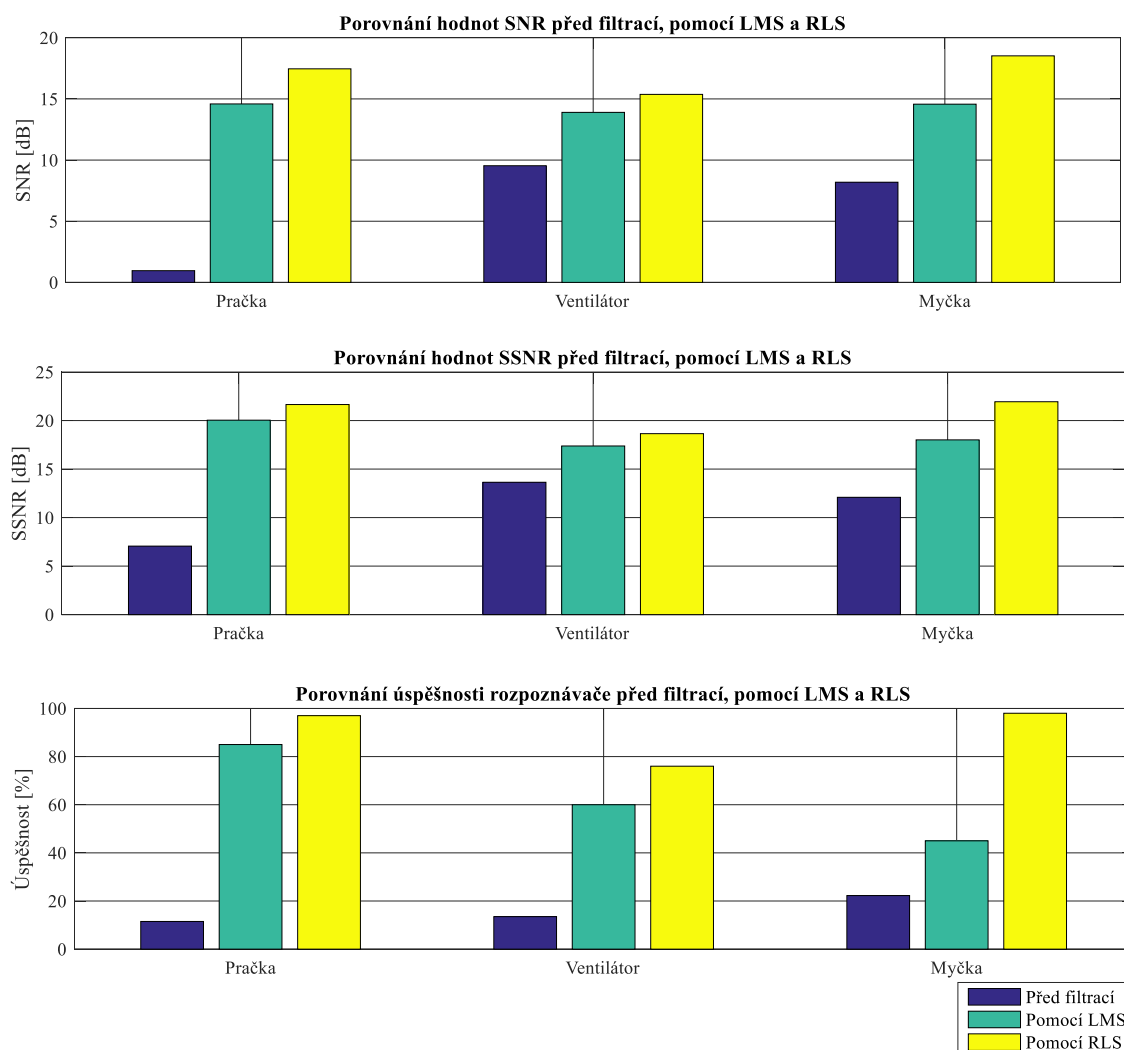
Obr. 39: Časové průběhy před a po filtraci za použití RLS algoritmu.



Obr. 40: Testované signály zobrazené v časově-frekvenční oblasti před a po filtraci pomocí RLS algoritmu znázorněné ve spektrogramu.

Následující grafy (Obr. 41) zobrazují porovnání výsledků filtrace pro oba použité algoritmy. V horním grafu se nachází průměrné hodnoty  $SNR$  pro všechny nahrávky před filtrací (modře), při použití LMS algoritmu (zeleně) a pomocí RLS algoritmu (žlutě). Druhý graf zobrazuje průměrné hodnoty  $SSNR$  pro nahrávky před filtrací (modře), při použití LMS algoritmu (zeleně) a pro RLS algoritmus (žlutě). V posledním grafu je zobrazena úspěšnost rozpoznávací WSR před filtrací (modře), pomocí filtrace při použití LMS (zeleně) a pomocí RLS (žlutě).

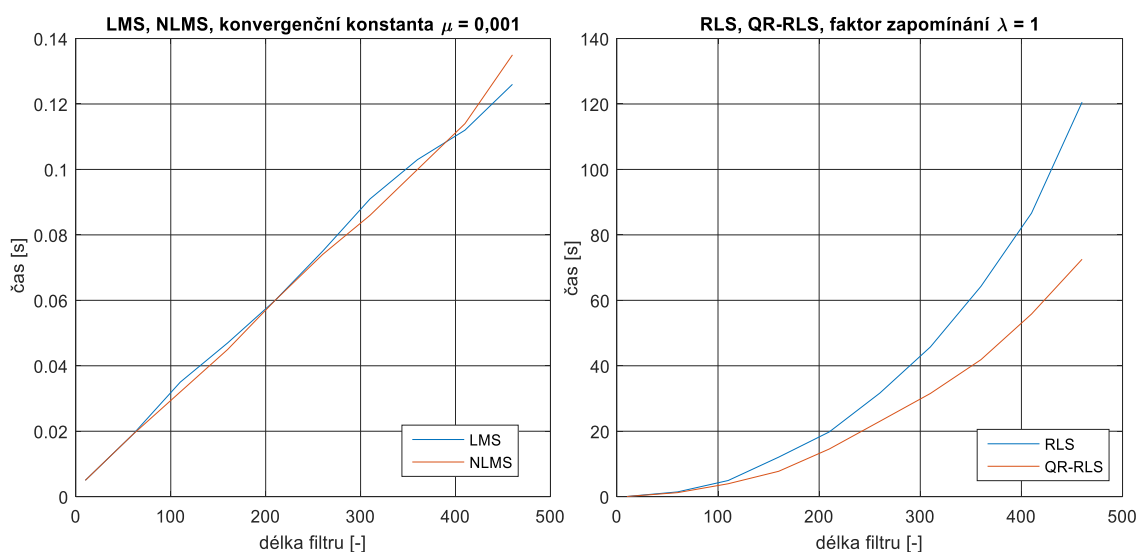
V jednotlivých grafech lze pozorovat, že nejlepších výsledků dosahujeme při použití RLS algoritmu. Algoritmus RLS se tedy jeví jako lepší, jelikož dosahuje lepších výsledků při porovnání s LMS. To je důvodem proč RLS metoda vykazuje výbornou kvalitu zvuku.



Obr. 41: Porovnání výsledků filtrace pro algoritmy LMS a RLS.

Ovšem u této metody byl stanoven řád filtru na hodnotu 200 pro všechna testování. V tomto případě se nejednalo o nejideálnější nastavení filtru z hlediska hodnot  $SNR$ . Ideální hodnoty řádu filtru se pohybovali přibližně v rozmezí od 500 do 600. Kdyby byla použita ideální hodnota řádu filtru, byla by výpočetní doba pro jeden příkaz přibližně 250 sekund. Pro použití v reálných aplikacích by takové doby výpočtu nebyly vhodné. Při nastavení řádu filtru do hodnoty 50, se doby výpočtu pohybují v okolí 1 sekundy.

Pokud bude srovnána matematická náročnost obou těchto algoritmů, lze vidět, že algoritmus RLS, platí za svoji přesnost a rychlost adaptace svojí daň. Jedná se totiž o velmi matematicky náročný algoritmus a tato náročnost je zobrazena v následujícím grafu (Obr. 42). Zatímco u LMS metody se doby výpočtu pohybují v řádech desetin sekund, tak u RLS algoritmu se již pohybujeme v řádech desítek sekund. Doby výpočtu se odvíjejí od hardwaru zařízení, na kterém adaptivní filtry pracují.

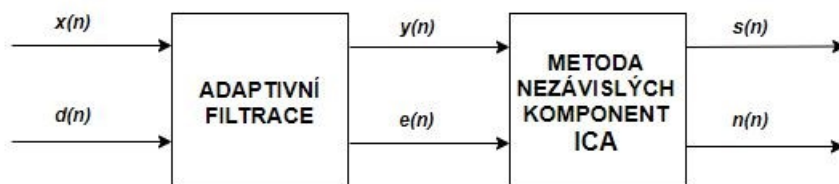


Obr. 42: Výpočetní doba adaptivních filtrací.

#### 4.7.2 Kombinace adaptivní filtrace a ICA

Bylo zjištěno, že výsledky LMS algoritmu nebyly příliš vyhovující, co se týká úspěšnosti rozpoznávače řeči. Proto byla použita jiná metoda, která se vyznačuje lepšími vlastnostmi, téměř ideálními pro kvalitu zvuku. Jedná se o matematicky náročnou metodu RLS. S použitím této metody byla zvýšena úspěšnost rozpoznávače téměř ve všech případech nad 90 %. Tato matematicky náročná metoda, ale způsobila příliš vysokou dobu výpočtu. V reálných aplikacích by to znamenalo, že by uživatel musel čekat příliš dlouhou dobu na vyhodnocení filtrace a prodleva mezi vyřčením fráze a jejím vyhodnocením by trvala až moc dlouho dobu.

Z toho důvodu byla použita jiná metoda. Jedná se o metodu nezávislých komponent ICA, která nepracuje na úrovni filtrace signálů. Jedná se o metodu, která slouží k separaci dvou a více signálů od sebe. Tato metoda nebyla použita jako samotná, ale jen v kombinaci s adaptivní filtrací. Jelikož nebyly splněny všechny podmínky při nahrávání všech frází pro možnost použití této metody jako samotné.



Obr. 43: Kombinace AF a ICA.

Na Obr. 43 je znázorněno, jakým způsobem je tato kombinace řešena. Hodnota  $x(n)$  označuje vstupní signál kontaminovaný šumem,  $d(n)$  signál šumu,  $y(n)$  představuje odfiltrovanou řeč pomocí AF a  $e(n)$  je chybový signál neboli chyba filtrace. Právě hodnoty, které jsou zpracované pomocí AF jsou dále zpracovávány pomocí metody nezávislých komponent. Dříve nemohla být tato metoda použita, jelikož jsme neměli k dispozici řeč z více míst, na které by bylo možné pozorovat zkreslení způsobené prostředím. Pomocí filtrace dostaneme dva signály, kdy první signál  $y(n)$ , představuje odfiltrovanou řeč, ve které se ale stále nachází i jistá úroveň šumu. Druhým signálem  $e(n)$  je vyjádřen odfiltrovaný šum, ve kterém se nachází i určitá úroveň odfiltrované řeči. Za tohoto předpokladu může být použita metoda ICA, jelikož jsou signály řeči a šumu zastoupeny s jistým zkreslením v obou signálech. Pomocí této separace získáme signál  $s(n)$  a šum  $n(n)$ . Hodnoty porovnání budou stanoveny pouze podle úspěšnosti filtrace, jelikož metoda ICA mění amplitudu a pořadí jednotlivých nahrávek.

Následující Tab. 10: zobrazuje úspěšnost rozpoznávače řeči WSR. Nenachází se zde hodnoty  $SNR$ , jak již bylo řečeno, ICA mění parametry amplitudy a hodnoty  $SRN$  by se tedy nedaly spočítat v přímém poměru k originálnímu signálu. Tabulka je rozdělena na dvě části. První část zobrazuje úspěšnost rozpoznávače řeči při použití metody LMS. Ve druhé části jsou uvedeny hodnoty právě pro danou kombinaci adaptivního filtru a metody nezávislých komponent. Zatímco filtrace pomocí LMS dosahovala dobrých výsledků rozpoznání pouze pro šum pračky, pomocí této kombinace byla zvýšena úspěšnost rozpoznání u všech použitých šumů.

Tab. 10: Úspěšnost rozpoznávače řeči pro LMS a LMS + ICA.

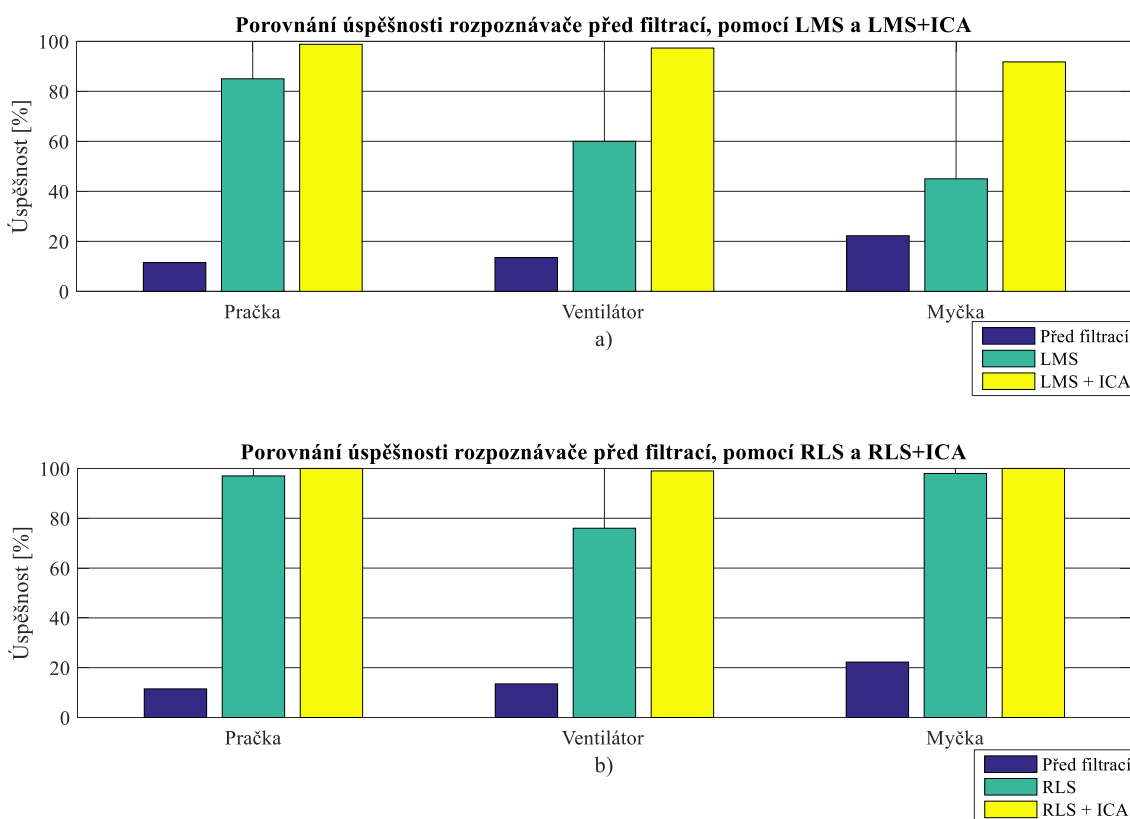
Nahrávky	Úspěšnost LMS [%]			Úspěšnost LMS +ICA [%]		
	Pračka	Ventilátor	Myčka	Pračka	Ventilátor	Myčka
Light on	95	89	89	99	100	98
Light off	96	28	25	100	96	55
Blinds up	98	25	5	98	98	98
Blinds down	97	82	28	100	100	97
Volume up	82	79	34	98	99	99
Volume down	99	72	65	100	98	100
Increase temperature	25	15	15	97	88	87
Decrease temperature	90	97	99	99	100	100

Tab. 11: Úspěšnost rozpoznávače řeči pro RLS a RLS + ICA.

Nahrávky	Úspěšnost RLS [%]			Úspěšnost RLS +ICA [%]		
	Pračka	Ventilátor	Myčka	Pračka	Ventilátor	Myčka
Light on	99	100	100	100	100	100
Light off	97	13	98	100	98	100
Blinds up	94	100	100	100	100	100
Blinds down	99	89	98	100	99	100
Volume up	97	98	99	100	100	100
Volume down	96	100	100	100	100	100
Increase temperature	94	22	97	100	98	100
Decrease temperature	100	89	99	100	99	100

Podle výsledků došlo k výraznému zlepšení úspěšnosti rozpoznávače při použití kombinace AF+ICA. Dosažení ještě lepších výsledků úspěšnosti, bylo dosaženo pomocí kombinace AF + ICA, kdy byl místo LMS použit RLS algoritmus. Ovšem doby výpočtu nejsou vhodné pro reálné použití. V Tab. 11 je zobrazena úspěšnost pro RLS i při použití kombinace RLS + ICA při nastavené délce filtru na 200 a faktoru zapomínání  $\lambda = 1$ .

V grafech na Obr. 44 je zobrazena úspěšnost rozpoznávače řeči pro metody LMS a RLS. Při použití LMS algoritmu byla vyšší přesnost zajištěna pouze pro filtraci hluku pračky. Hluk ventilátoru a myčky se tomuto algoritmu nedařilo příliš úspěšně filtrovat. Proto byla i výsledná úspěšnost rozpoznávače řeči nedostačující. Pro zlepšení výsledků, byla použita kombinace dvou metod současně. Jednalo se o kombinaci metody adaptivní filtrace a metody nezávislých komponent. Na Obr. 44 a) je zobrazena kombinace LMS a ICA. Při této kombinaci bylo dosaženo výsledků úspěšnosti 99 % pro pračku, 97 % pro hluk ventilátoru a 92 % pro hluk myčky. Zatímco při použití samotné metody LMS byla úspěšnost 85 % pro pračku, 60 % pro ventilátor a 45 % pro myčku. Na Obr. 44 b) je zobrazena kombinace RLS a ICA. Při použití samotné metody RLS byla úspěšnost rozpoznávače velmi dobrá. Při kombinaci s metodou ICA se úspěšnost zvýšila natolik že dosahovala 100 % ve všech případech.

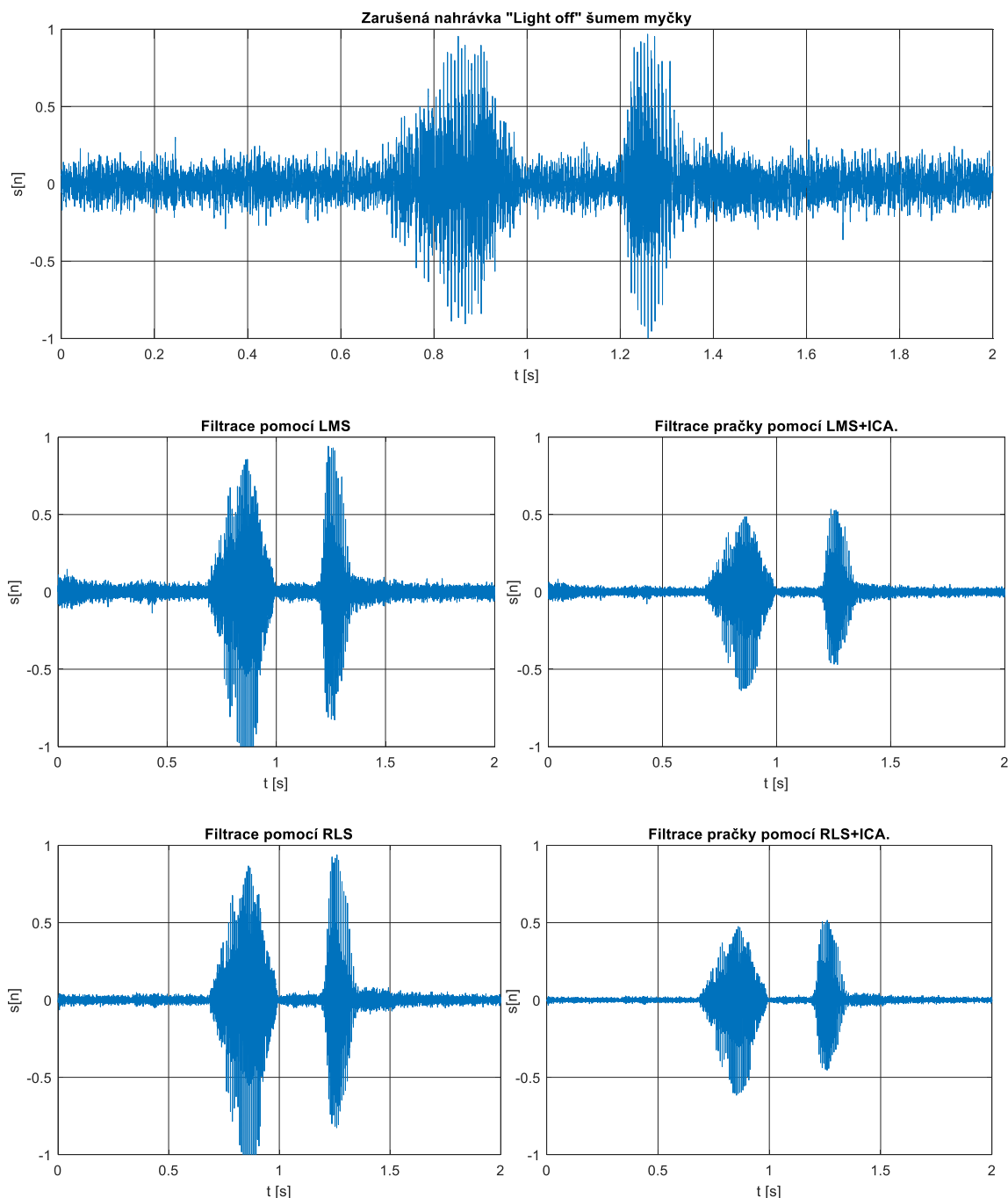


Obr. 44: Grafické zobrazení úspěšnosti rozpoznávače pro: a) kombinaci LMS a ICA, b) kombinaci RLS a ICA.

Na Obr. 45 lze pozorovat výsledky filtrace pro metody LMS, RLS a jejich kombinace s metodou nezávislých komponent ICA. V grafech je zobrazen pozvolný pokles úrovně šumu u algoritmu LMS. U algoritmu RLS již není tento pokles viditelný, jelikož se metoda vyznačuje rychlou



adaptací na signál. V případě uskutečnění poslechových testů, by bylo možné slyšet pozvolný pokles hluku v pozadí u metody LMS. Tato metoda tedy potřebuje velký počet iterací, aby se přiblížil k optimálnímu stavu filtrace na rozdíl od RLS, který pracuje velmi rychle. Při porovnání výsledků filtrace v kombinaci adaptivního filtru a metody ICA lze pozorovat výrazný pokles amplitudy. Z tohoto důvodu nebylo možné počítat hodnoty SNR při této kombinaci. Snížením amplitudy signálu je dosaženo i výrazného poklesu hluku pozadí. Pomocí metody ICA, která měla za důsledek snížení hluku pozadí, bylo dosaženo výrazného zlepšení úspěšnosti rozpoznávání jednotlivých výrazů.



Obr. 45: Porovnání výsledků filtrace pro LMS, RLS a jejich kombinace s metodu nezávislých komponent ICA.



### 4.7.3 Použití v automobilu

Při posouzení kvality filtrace v automobilu bylo vybráno 9 příkazů, pomocí kterých může uživatel ovládat osobní automobil. Zde jsou zobrazeny použité příkazy:

- Accept call.
- Decline call.
- Auto lights.
- Cruise control one hundred.
- Cruise control one hundred thirty.
- I stop on.
- I stop off.
- Radio on.
- Radio off.

Testování bylo prováděno v automobilu poháněném elektřinou. Měření probíhalo v následujících rychlostech 20 km/h, 50 km/h, 80 km/h, 100 km/h a 130 km/h při zavřených oknech a při otevřeném levém zadním okně. Diplomová práce obsahuje výsledky pouze pro rychlosti 80 km/h, 100 km/h a 130 km/h. Výsledky pro zavřená okna jsou zobrazeny v tabulkách (Tab. 12, Tab. 13, Tab. 14) a pro otevřené okno jsou výsledky zobrazeny v Tab. 15, Tab. 16, Tab. 17. Při nižších rychlostech nebylo třeba použití filtrace a úspěšnost rozpoznávače byla výborná. Výsledky filtrace se jsou uvedeny pro algoritmus LMS. Tabulky jsou rozděleny na tři části. V prvním sloupci je zobrazeno ideální nastavení řádu filtru podle hodnot  $SNR$ . Druhá část zobrazuje výsledky  $SNR$ ,  $SSNR$  a úspěšnosti rozpoznávače před filtrací a ve třetí části jsou zobrazeny stejné výsledky po filtraci pomocí LMS algoritmu.

Tab. 12: Výsledky filtrace a rozpoznávače pro rychlost 80 km/h, zavřená okna (LMS).

Filtrace LMS	Výsledky při rychlosti 80 km/h, $\mu = 0,001$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Accept call	290	-1,92	3,90	98	16,57	19,94	99
Decline call	350	2,13	6,84	99	20,44	23,77	99
Auto lights	410	-0,04	7,10	99	18,29	22,30	100
Cruise control one hundred	290	-3,80	0,30	100	14,39	16,26	100
Cruise control one hundred thirty	150	-5,39	-0,38	93	12,78	14,41	88
I stop on	160	-7,97	-2,03	75	10,68	13,91	97
I stop off	190	-6,13	1,14	98	12,32	16,03	51
Radio on	290	-6,18	-1,02	25	12,08	15,23	78
Radio off	150	-3,95	0,42	85	13,99	16,33	24

Tab. 13: Výsledky filtrace a rozpoznávače pro rychlost 100 km/h, zavřená okna (LMS).

Filtrace LMS	Výsledky při rychlosti 100 km/h, $\mu = 0,001$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Accept call	60	-3,97	2,98	79	12,37	15,65	98
Decline call	90	0,09	5,50	99	16,30	19,34	100
Auto lights	30	-2,09	3,26	99	14,21	18,31	98
Cruise control one hundred	80	-6,48	-2,73	99	10,30	12,51	99
Cruise control one hundred thirty	80	-8,07	-4,01	91	8,81	10,98	100
I stop on	80	-10,02	-4,08	21	7,05	10,53	89
I stop off	20	-8,18	-2,11	79	8,47	12,29	41
Radio on	30	-8,22	-3,54	15	8,29	11,30	81
Radio off	30	-6,01	-1,09	98	10,04	12,11	83

Tab. 14: Výsledky filtrace a rozpoznávače pro rychlost 130 km/h, zavřená okna (LMS).

Filtrace LMS	Výsledky při rychlosti 130 km/h, $\mu = 0,001$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Accept call	140	-7,11	-2,24	75	10,87	14,82	68
Decline call	120	-3,06	0,72	99	14,79	18,08	100
Auto lights	120	-5,23	-0,02	12	12,68	17,54	92
Cruise control one hundred	130	-9,09	-6,56	95	8,96	11,24	92
Cruise control one hundred thirty	140	-10,68	-7,48	98	7,54	9,65	100
I stop on	190	-13,16	-7,67	9	5,84	9,24	26
I stop off	200	-11,32	-5,78	97	7,22	10,93	89
Radio on	120	-11,37	-7,44	5	7,01	10,34	75
Radio off	120	-9,14	-5,31	99	8,65	11,82	82

Tab. 15: Výsledky filtrace a rozpoznávače pro rychlost 80 km/h, otevřené okno (LMS).

Filtrace LMS	Výsledky při rychlosti 80 km/h, $\mu = 0,1$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Accept call	1250	-36,13	-32,01	85	9,92	16,20	100
Decline call	1050	-32,08	-28,15	94	13,63	18,74	100
Auto lights	990	-34,25	-29,36	92	11,63	17,85	100
Cruise control one hundred	1280	-37,87	-34,13	3	8,37	13,99	100
Cruise control one hundred thirty	1250	-39,46	-36,76	6	7,05	12,01	100
I stop on	1200	-42,19	-36,44	25	4,92	11,56	100
I stop off	1320	-40,35	-35,04	88	6,32	13,14	15
Radio on	1300	-40,39	-35,45	19	6,09	11,78	99
Radio off	1350	-38,16	-34,67	79	7,67	13,85	9

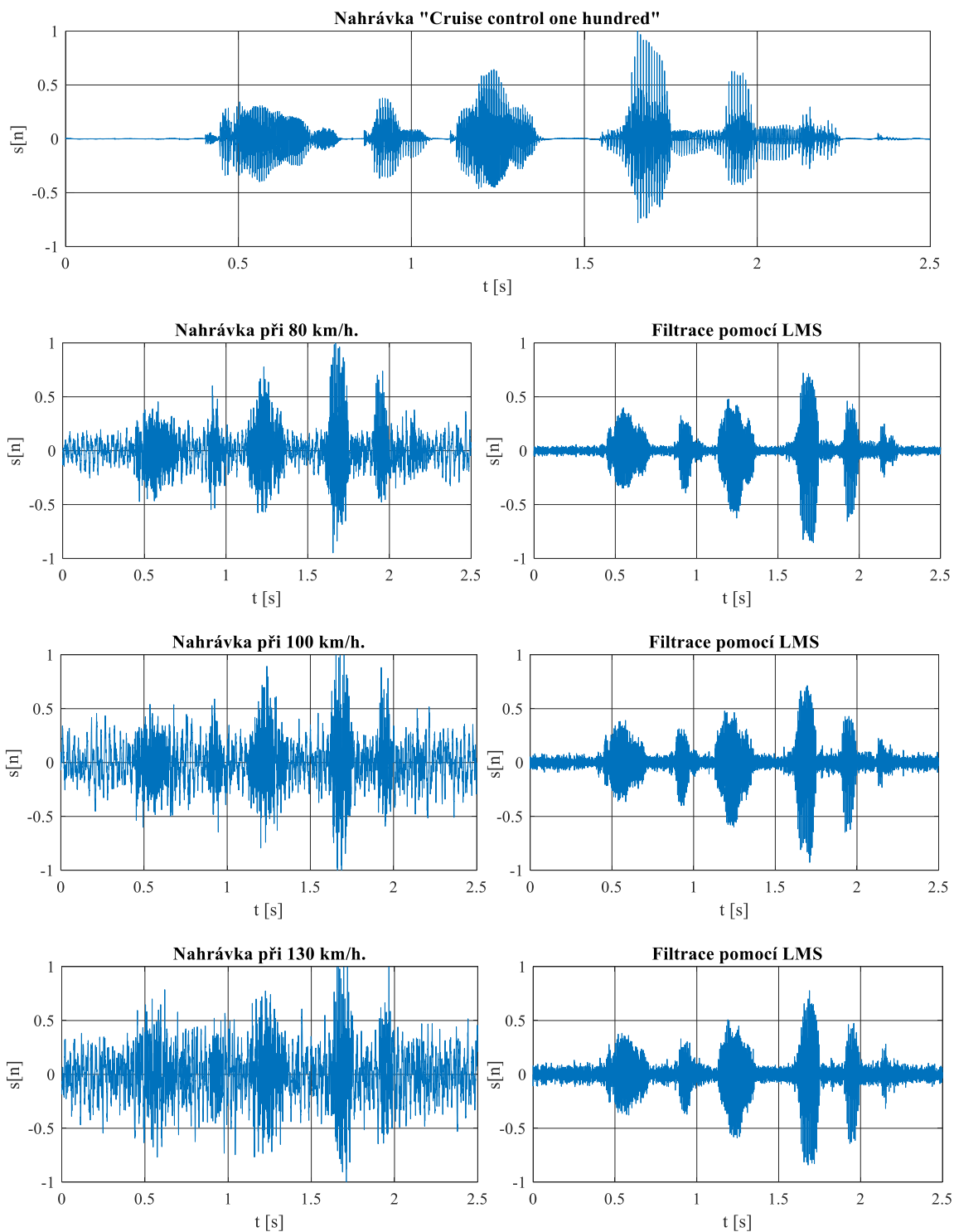
Tab. 16: Výsledky filtrace a rozpoznávače pro rychlost 100 km/h, otevřené okno (LMS).

Filtrace LMS	Výsledky při rychlosti 100 km/h, $\mu = 0,1$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Accept call	940	-33,26	-32,58	85	9,13	15,26	98
Decline call	800	-34,26	-30,71	95	12,10	17,89	100
Auto lights	880	-31,69	-29,18	15	10,89	16,45	99
Cruise control one hundred	800	-38,22	-35,46	18	8,03	12,33	100
Cruise control one hundred thirty	980	-40,12	-37,19	15	6,89	11,20	92
I stop on	880	-42,59	-38,61	68	4,91	10,56	99
I stop off	980	-41,05	-36,47	57	5,96	12,11	15
Radio on	880	-40,87	-36,54	28	5,59	11,94	100
Radio off	30	-39,12	-35,53	46	6,27	11,58	4

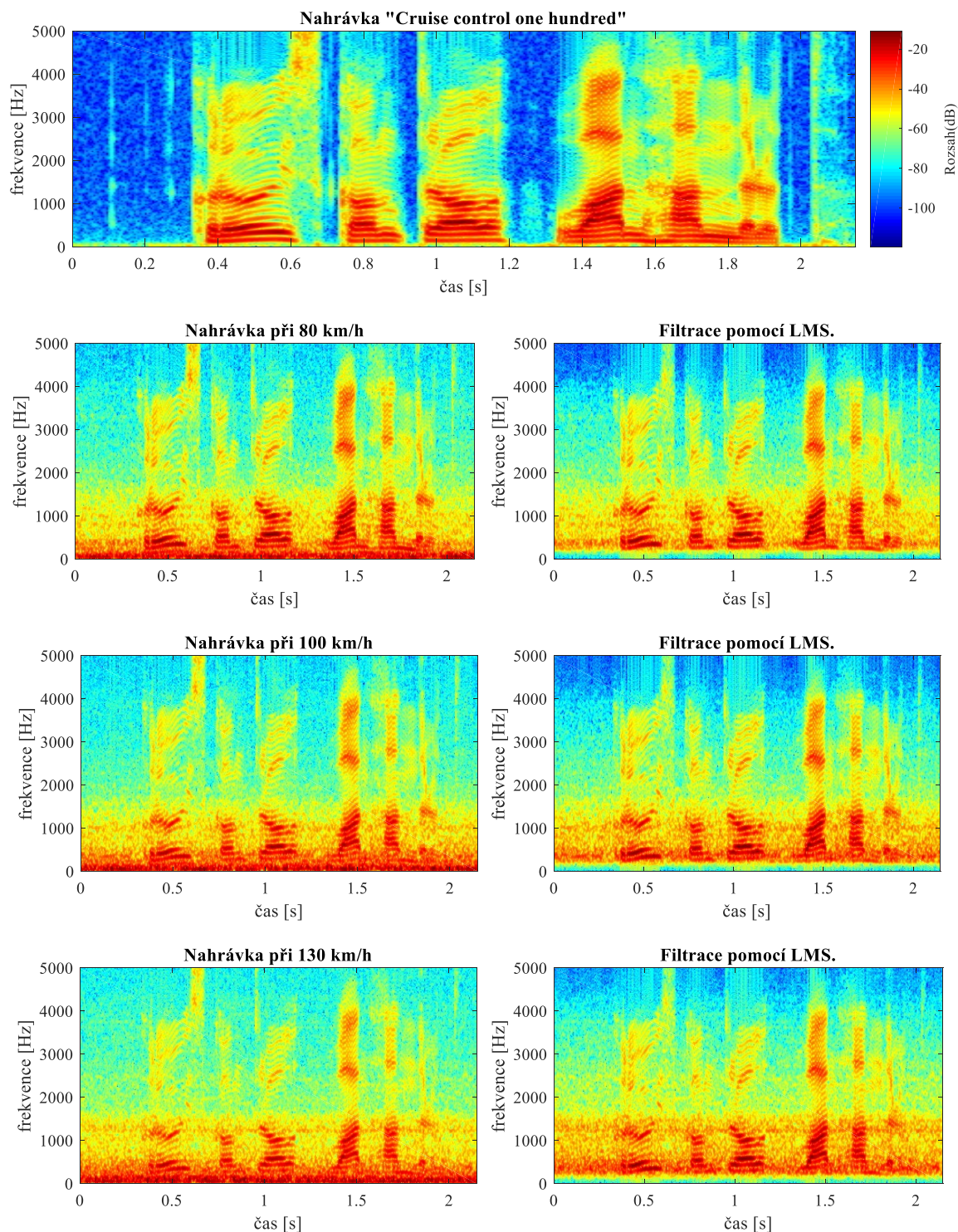
Tab. 17: Výsledky filtrace a rozpoznávače pro rychlost 130 km/h, otevřené okno (LMS).

Filtrace LMS	Výsledky při rychlosti 130 km/h, $\mu = 0,1$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
<b>Accept call</b>	770	-32,17	-20,70	80	7,57	11,80	98
<b>Decline call</b>	750	-28,12	-19,02	93	11,19	14,55	99
<b>Auto lights</b>	730	-30,29	-19,24	1	9,20	14,05	99
<b>Cruise control one hundred</b>	690	-33,43	-22,61	0	5,94	8,92	98
<b>Cruise control one hundred thirty</b>	700	-35,02	-25,41	0	4,83	7,91	99
<b>I stop on</b>	890	-38,22	-27,79	1	3,47	7,66	99
<b>I stop off</b>	900	-36,38	-28,12	87	4,58	9,06	8
<b>Radio on</b>	760	-36,43	-26,33	1	4,29	7,72	98
<b>Radio off</b>	700	-34,20	-24,81	54	5,63	9,31	7

Jak již bylo řečeno, úspěšnost filtrace byla testována v pěti různých rychlostech, kdy výsledky jsou uvedeny pouze pro tři z nich. Jelikož je použitý rozpoznávač velice robustní a obsahuje obrovskou slovní zásobu různých řečníků pracuje rozpoznávač pro nízké rychlosti bez potřeby filtrace. Z toho důvodu jsou výsledky zpracované jen pro rychlosti, ve kterých bylo třeba použití adaptivní filtrace za účelem zlepšení úspěšnosti rozpoznávače řeči. Pro každou testovanou nahrávku byla úspěšnost rozpoznávání testována stokrát. V tabulkách je zobrazena úspěšnost filtrace z hlediska SNR a rozpoznání WSR. Se zvyšující se rychlostí lze pozorovat, že odstup signálu od šumu klesá. Pro rychlost 80 km/h je hodnota SNR před filtrací -3,7 dB, pro 100 km/h je SNR -5,9 a pro 130 km/h je tato hodnota rovna téměř -9 dB. Hodnoty SSNR před filtrací jsou pro 80 km/h 1,8 dB, pro 100 km/h -0,6 dB a pro 130 km/h je tato hodnota rovna -4,64 dB. Stejný fakt platí i pro hodnoty úspěšnosti rozpoznávače řeči. Se zvyšující se rychlostí dochází ke snižování úspěšnosti rozpoznávání jednotlivých příkazů. Pro rychlost 80 km/h byla úspěšnost 85 %. Poté s každou další rychlostí klesala úspěšnost o 10 % až na konečných 65 %. Po filtraci, došlo při rychlosti 80 km/h k mírnému zhoršení na 82 %. Pro rychlost 100 km/h byla úspěšnost po filtraci 87 % a pro 130 km/h byla úspěšnost 81 %. Při otevřeném oknu již klesla hodnota SNR na rozmezí od -27 dB do -31 dB. Pro jednotlivé rychlosti. Hodnoty SSNR se pohybují v rozmezí od -28 dB do -20 dB. Při použití filtrace byly hodnoty SNR zvýšeny na 7 dB pro 80 km/h, 6,3 dB pro 100 km/h a 5 dB pro 130 km/h. Hodnoty SSNR byly zvýšeny na 11,7 dB, 10,8 dB a 8,3 dB při stoupající rychlosti. Úspěšnost rozpoznávání jednotlivých výrazů klesala se zvyšující se rychlostí. Pro rychlost 80 km/h byla úspěšnost 45 %, pro 100 km/h 38 % a pro 130 km/h byla úspěšnost 29 % při otevřeném oknu. Filtrací pomocí LMS algoritmu byla zvýšena úspěšnost rozpoznávače na 65 % u všech rychlostí.



Obr. 46: Časové průběhy před a po filtraci za použití LMS algoritmu.



Obr. 47: Testované signály zobrazené v časově-frekvenční oblasti před a po filtraci pomocí LMS algoritmu znázorněné ve spektrogramu.

Obr. 46 zobrazuje výsledek filtrace pomocí LMS algoritmu. V grafech se nachází nahrávka „Cruise control one hundred“, ve všech rychlostech, které byly použity. Levá část zobrazuje kontaminaci nahrávky s hlukem v dané rychlosti. Pravá část zobrazuje časové průběhy po filtraci za použití LMS algoritmu. Na Obr. 47 se nachází výsledky filtrace v časově-frekvenční oblasti. S rostoucí rychlostí lze v grafech pozorovat zvyšování úrovně hluku v pozadí řečové nahrávky. Podle hodnocení

z hlediska úspěšnosti rozpoznávače řeči lze pozorovat, že docházelo ke zhoršení rozpoznávání nahrávek „*I stop off*“ a „*Radio off*“. Je to způsobeno tím, že energie slova „*off*“ není příliš velká. Z toho důvodu se neznělá hláska „*f*“ při filtraci v zarušené nahrávce částečně vytratí. Pro rozpoznávač už je složité detekovat celou tuto frázi správně. Místo správných nahrávek docházelo k rozpoznávání slov „*I stop on*“ a „*Radio on*“. Při provedení poslechové metody byla tato hláska plně slyšitelná, ale pro rozpoznávač již nebylo možné detekovat tuto frázi ve správném znění.

Následně byla provedena filtrace za použití druhého algoritmu, a to RLS. U tohoto algoritmu byla stanovena hodnota faktoru zapomínání  $\lambda = 1$ . Hodnoty řádů filtru se pohybují v rozsahu od 10 do 70. Z časového hlediska to představuje doby výpočtu do 3 sekund. Oproti použití v domácnosti, kdy byla hodnota řádu filtru 200 (19 sekund) by mohlo být toto použití využito v reálných aplikacích.

V Tab. 18, Tab. 19, Tab. 20 jsou uvedeny výsledky pro filtraci pomocí RLS algoritmu při zavřených oknech. V Tab. 21, Tab. 22, Tab. 23 jsou zobrazeny výsledky pro otevřené okno. Jsou zde zobrazeny hodnoty před filtrací a po filtraci. Každá nahrávka byla testována celkem stokrát. Pro rychlost 80 km/h byla úspěšnost rozpoznávače zlepšena na 98 %. Při rychlosti 100 km/h dosahovala úspěšnost 90 % a pro 130 km/h byla výsledná úspěšnost 84 %. Úspěšnost při vyšších rychlostech byla snížena zejména z důvodu, že se rozpoznávači nedařilo správně určit nahrávku „*I stop on*“. Druhým výsledkem z tabulek je úspěšnost filtrace zobrazená pomocí hodnot *SNR* a *SSNR*. Průměrné zlepšení pro hodnoty *SNR* se pohybovalo v rozmezí od 16,5 dB do 18,2 dB. Co se týká hodnot *SSNR* byl rozsah zlepšení těchto hodnot od 14,2 dB do 17 dB.

Při otevřeném oknu byla úspěšnost rozpoznávače řeči zvýšena na 69 % pro rychlost 80 km/h, pro rychlost 100 km/h byla úspěšnost 62 % a pro 130 km/h byla konečná úspěšnost 60 %. Nedošlo zde tedy k větší úspěšnosti než při použití LMS algoritmu z toho důvodu, že se rozpoznávači nedařilo správně rozeznat nahrávky „*I stop off*“ a „*Radio off*“. Pro jednotlivé rychlosti byly hodnoty *SNR* zvýšeny na 8,7 dB, 6,5 dB a 5,2 dB a hodnoty *SSNR* na 11,7 dB, 9,4 dB a 7,5 dB. Tyto hodnoty se rovnají i výsledkům při použití LMS algoritmu.

Tab. 18: Výsledky filtrace a rozpoznávače pro rychlost 80 km/h, zavřená okna (RLS).

Filtrace RLS	Výsledky při rychlosti 80 km/h, $\lambda = 1$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Accept call	10	-1,92	3,90	98	16,36	19,78	99
Decline call	10	2,13	6,84	99	20,07	23,15	100
Auto lights	10	-0,04	7,10	99	18,21	22,25	100
Cruise control one hundred	10	-3,80	0,3	100	14,29	16,07	99
Cruise control one hundred thirty	10	-5,39	-0,38	93	12,74	14,36	100
I stop on	80	-7,97	-2,03	75	10,65	13,85	98
I stop off	60	-6,13	1,14	98	12,30	15,95	99
Radio on	10	-6,18	-1,02	25	12,07	14,87	88
Radio off	70	-3,95	0,42	85	13,91	16,19	97

Tab. 19: Výsledky filtrace a rozpoznávače pro rychlost 100 km/h, zavřená okna (RLS).

Filtrace RLS	Výsledky při rychlosti 100 km/h, $\lambda = 1$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Accept call	10	-3,97	2,98	79	12,34	15,60	98
Decline call	10	0,09	5,50	99	16,25	19,25	99
Auto lights	10	-2,09	3,26	99	14,18	18,21	99
Cruise control one hundred	10	-6,48	-2,73	99	10,24	12,44	99
Cruise control one hundred thirty	70	-8,07	-4,01	91	8,77	10,66	100
I stop on	50	-10,02	-4,08	21	7,06	10,49	23
I stop off	10	-8,18	-2,11	79	8,45	12,22	97
Radio on	10	-8,22	-3,54	15	8,28	11,25	98
Radio off	60	-6,00	-1,09	98	9,97	11,91	98



Tab. 20: Výsledky filtrace a rozpoznávače pro rychlost 130 km/h, zavřená okna (RLS).

Filtrace RLS	Výsledky při rychlosti 130 km/h, $\lambda = 1$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Accept call	10	-7,11	-2,24	75	10,91	14,81	78
Decline call	10	-3,06	0,72	99	14,77	18,06	98
Auto lights	10	-5,23	-0,02	12	12,65	17,48	88
Cruise control one hundred	10	-9,09	-6,56	95	8,94	11,21	98
Cruise control one hundred thirty	10	-10,68	-7,48	98	7,51	9,20	99
I stop on	60	-13,16	-7,67	9	5,83	8,94	29
I stop off	10	-11,32	-5,78	97	7,19	10,59	82
Radio on	80	-11,37	-7,44	5	7,00	9,98	89
Radio off	60	-9,14	-5,31	99	8,65	11,51	92

Tab. 21: Výsledky filtrace a rozpoznávače pro rychlost 80 km/h, otevřená okna (RLS).

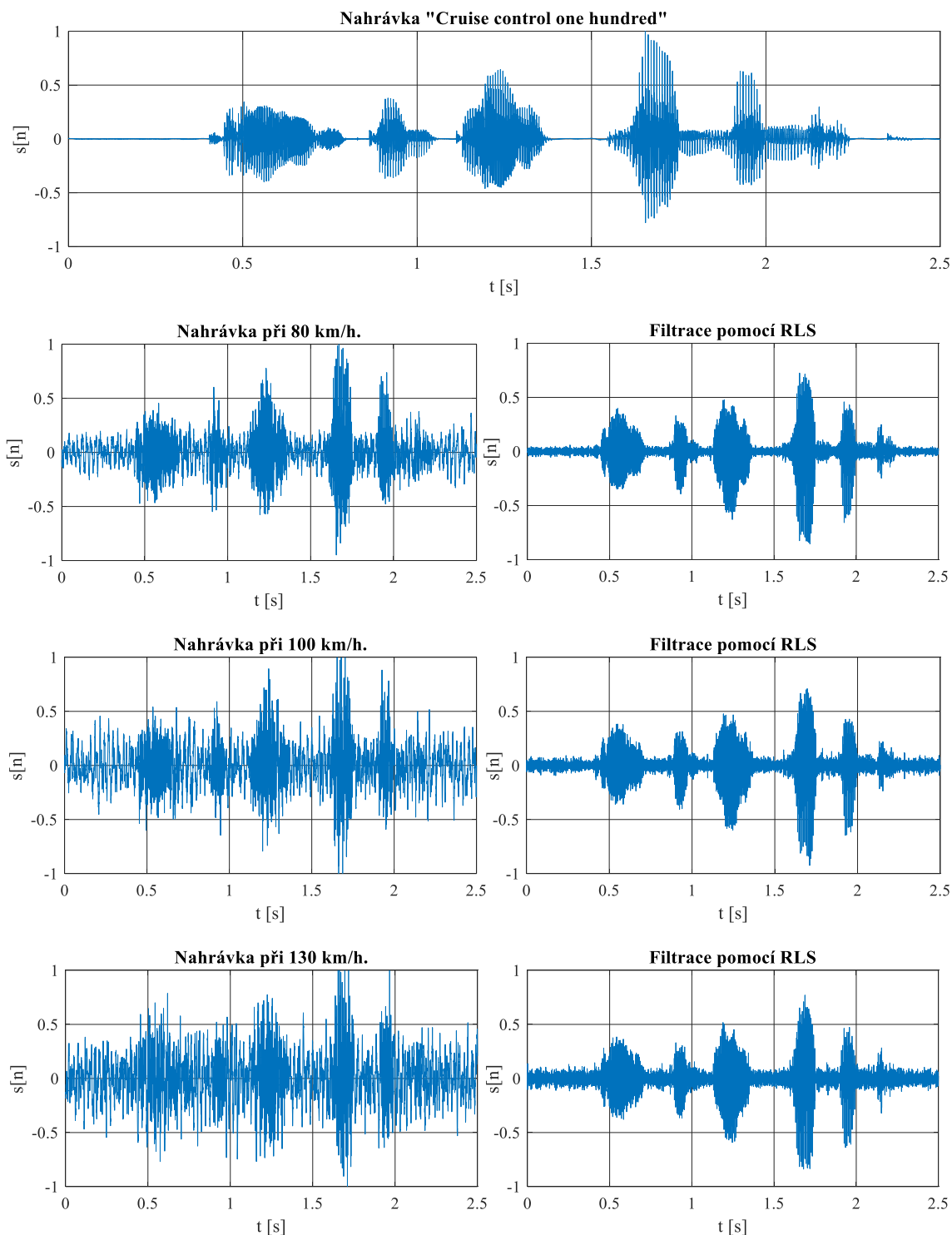
Filtrace RLS	Výsledky při rychlosti 80 km/h, $\lambda = 1$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Accept call	100	-36,13	-32,01	85	12,73	17,01	100
Decline call	110	-32,08	-28,15	94	13,55	18,97	100
Auto lights	110	-34,25	-29,36	92	14,52	19,06	100
Cruise control one hundred	90	-37,87	-34,13	3	10,82	12,63	100
Cruise control one hundred thirty	120	-39,46	-36,76	6	9,35	11,81	100
I stop on	100	-42,19	-36,44	25	7,24	11,51	55
I stop off	130	-40,35	-35,04	88	8,91	13,06	64
Radio on	120	-40,39	-35,45	19	8,59	11,07	88
Radio off	100	-38,16	-34,67	79	10,32	13,42	51

Tab. 22: Výsledky filtrace a rozpoznávače pro rychlost 100 km/h, otevřené okno (RLS).

Filtrace RLS	Výsledky při rychlosti 100 km/h, $\lambda = 1$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Accept call	180	-33,26	-32,58	85	9,52	13,35	100
Decline call	170	-34,26	-30,71	95	13,28	17,15	100
Auto lights	180	-31,69	-29,18	15	11,28	15,28	99
Cruise control one hundred	210	-38,22	-35,46	18	7,47	10,23	100
Cruise control one hundred thirty	180	-40,12	-37,19	15	6,27	9,88	100
I stop on	180	-42,59	-38,61	88	4,80	8,59	68
I stop off	200	-41,05	-36,47	57	5,95	9,54	8
Radio on	180	-40,87	-36,54	28	5,82	9,26	98
Radio off	60	-39,12	-35,53	46	7,36	10,26	7

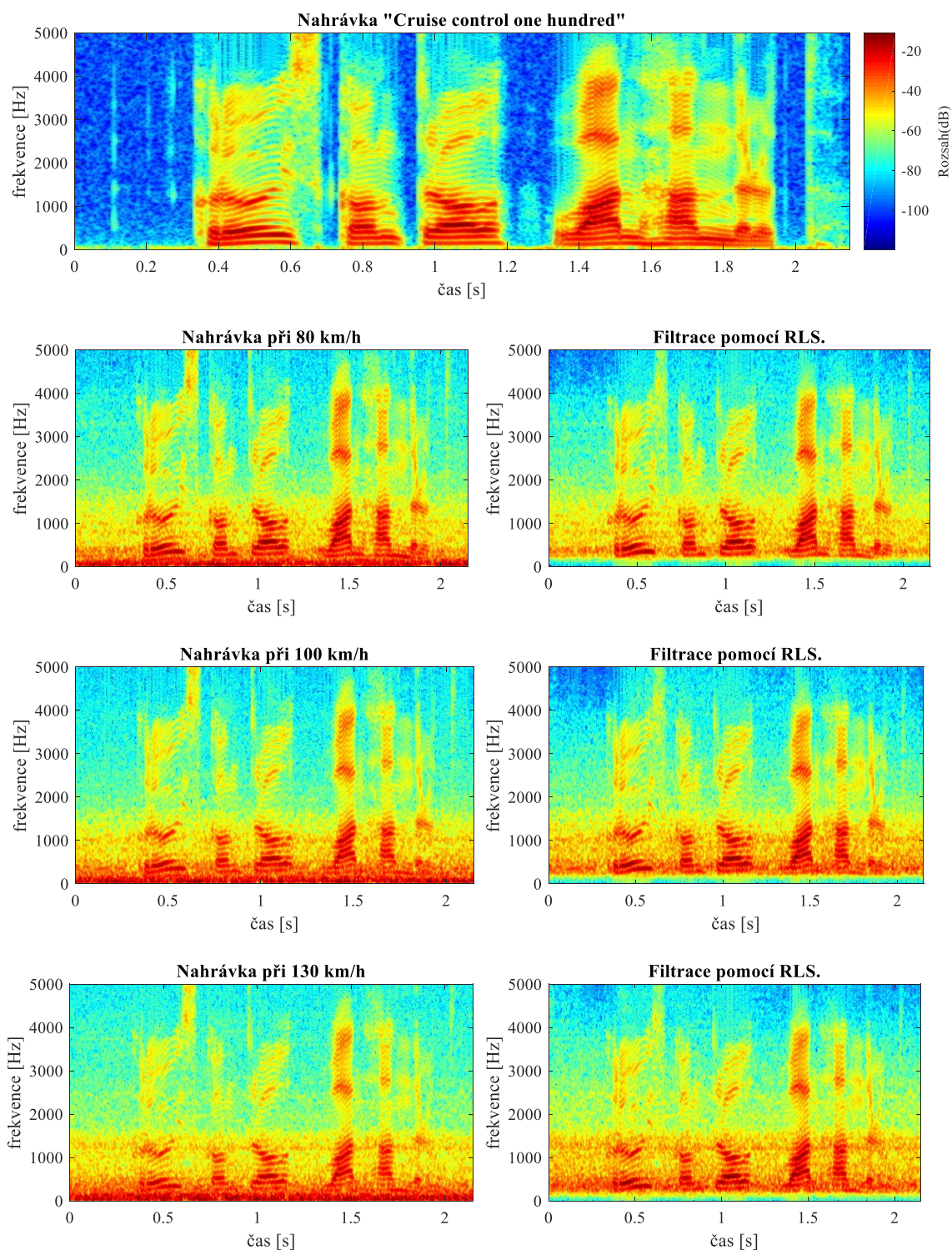
Tab. 23: Výsledky filtrace a rozpoznávače pro rychlost 130 km/h, otevřené okno (RLS).

Filtrace RLS	Výsledky při rychlosti 130 km/h, $\lambda = 1$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Accept call	210	-32,17	-20,70	80	7,64	10,82	98
Decline call	200	-28,12	-19,02	93	11,22	14,28	99
Auto lights	220	-30,29	-19,24	1	9,30	13,74	99
Cruise control one hundred	240	-33,43	-22,61	0	5,98	7,72	100
Cruise control one hundred thirty	200	-35,02	-25,41	0	4,85	7,22	100
I stop on	220	-38,22	-27,79	1	3,48	6,37	65
I stop off	230	-36,38	-28,12	87	4,60	7,95	5
Radio on	190	-36,43	-26,33	1	4,33	6,50	85
Radio off	210	-34,20	-24,81	54	5,70	8,08	4



Obr. 48: Časové průběhy před a po filtraci za použití RLS algoritmu.

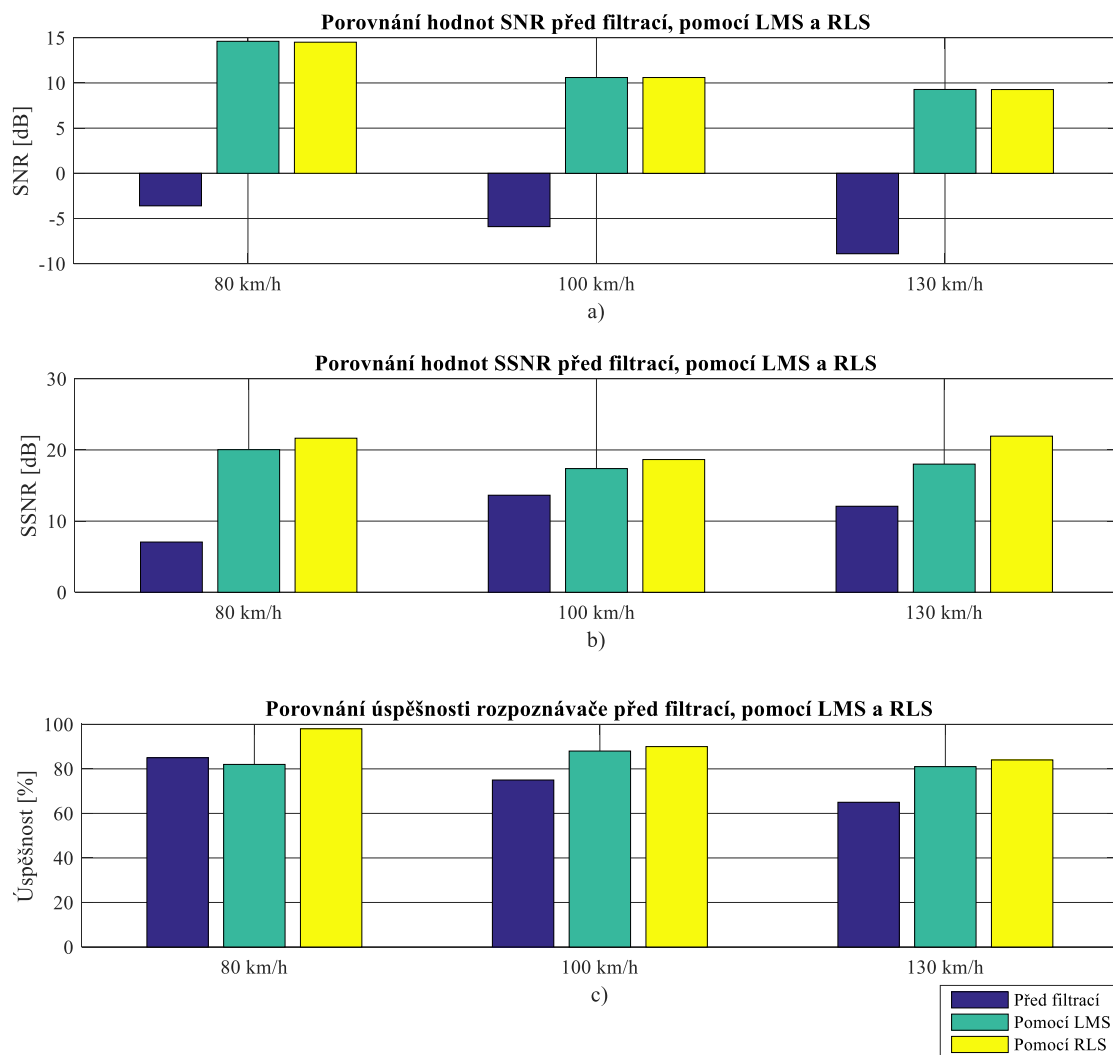
Obr. 48 zobrazuje výsledek filtrace pomocí RLS algoritmu. V grafech je zobrazena nahrávka „Cruise control one hundred“, která je postupně smíchána s hlukem v jednotlivých rychlostech. V pravé části je zobrazen výsledek filtrace. Na Obr. 49 se nachází výsledky filtrace zobrazené v časově-frekvenční oblasti. S rostoucí rychlostí lze v grafech pozorovat zvyšování úrovně hluku v pozadí řečové nahrávky.



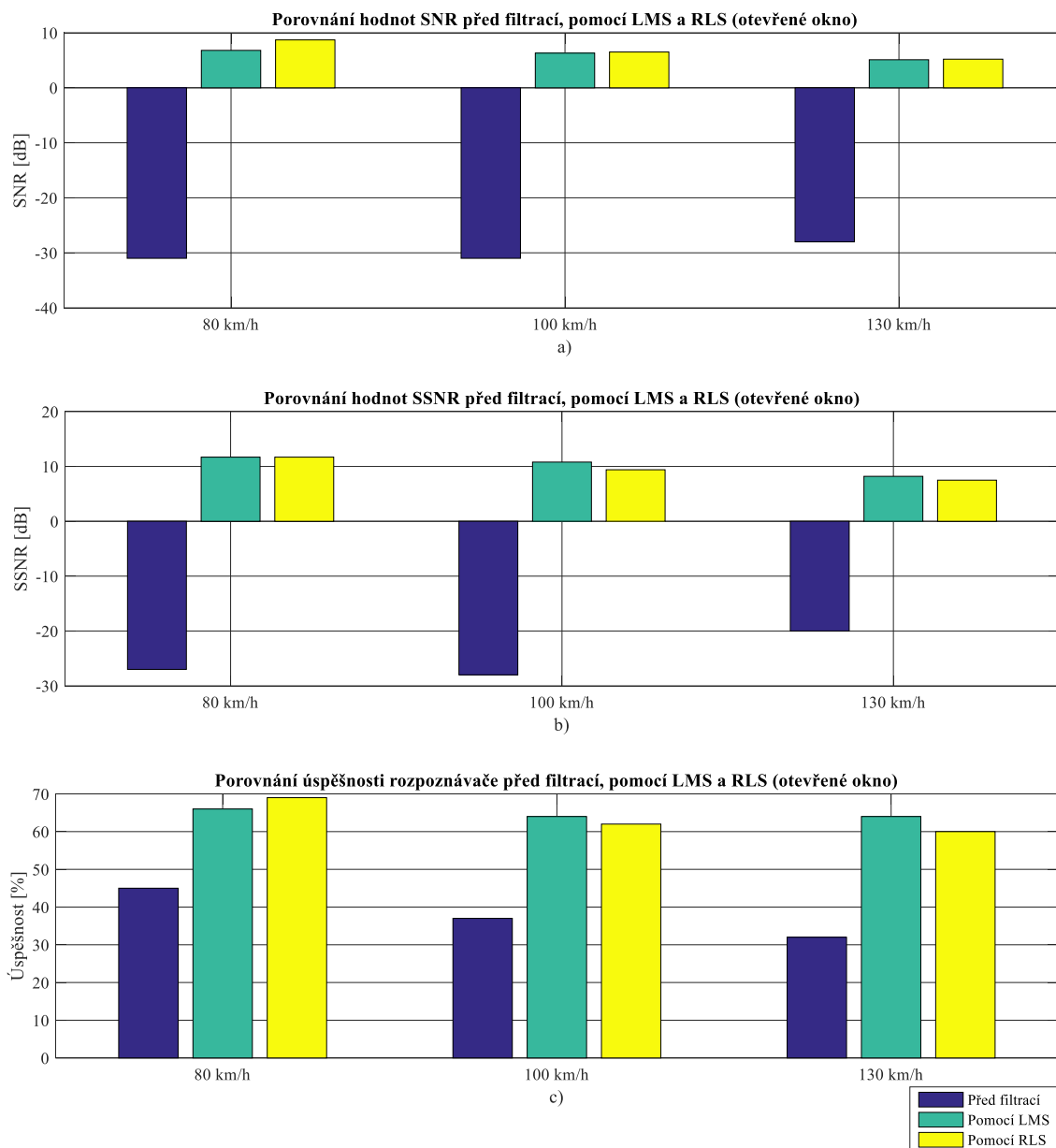
Obr. 49: Testované signály zobrazené v časově-frekvenční oblasti před a po filtraci pomocí RLS algoritmu znázorněné ve spektrogramu.

Následující grafy (Obr. 50, Obr. 51) zobrazují porovnání výsledků filtrace pro oba použité algoritmy, pro zavřená okna a pro otevřená okna. Graf a) zobrazuje průměrné hodnoty  $SNR$  všech použitých nahrávek před filtrací, za použití LMS algoritmu a pomocí RLS. V grafu b) jsou zobrazeny průměrné hodnoty  $SSNR$  a třetí graf c) zobrazuje úspěšnost rozpoznávače v jednotlivých rychlostech. V grafech při zavřených oknech lze pozorovat, že nejlepších výsledků dosahujeme při

použití RLS algoritmu. Algoritmus RLS se tedy jeví jako lepší, jelikož dosahuje lepších výsledků při porovnání s LMS. Zatímco v grafech při jednom otevřeném oknu, lze pozorovat, že lepších výsledků dosahujeme při použití LMS algoritmu.



Obr. 50: Porovnání výsledků filtrace při zavřených oknech pro: a) SNR, b) SSNR, c) Úspěšnost WSR, pro algoritmy LMS a RLS, použití v automobilu.



Obr. 51: Porovnání výsledků filtrace při otevřeném oknu pro: a) SNR, b) SSNR, c) Úspěšnost WSR, pro algoritmy LMS a RLS, použití v automobilu.

#### 4.7.4 Kombinace adaptivní filtrace a ICA

Stejně jako u domácnosti, byly i pro použití v automobilu zpracovány výsledky při kombinaci adaptivní filtrace a metody nezávislých komponent. V Tab. 24,

Tab. 25, Tab. 26, Tab. 27 je zobrazena úspěšnost rozpoznávací řeči. Tabulky jsou rozděleny na dvě části. První část zobrazuje hodnoty při použití samotné adaptivní filtrace. Ve druhé části jsou uvedeny výsledky pro kombinaci adaptivní filtrace a metody nezávislých komponent. První dvě tabulky zobrazují, že se úspěšnost rozpoznávání jednotlivých výrazů pohybovala nad 90 % kromě dvou nahrávek. Jednalo se o nahrávky „I stop off“ a „Radio off“. Jak již bylo řečeno, energie slova „off“ se při filtraci ztratí a rozpoznávač již nedokáže správně určit daný výraz. Stejných výsledků dosahujeme při použití této kombinace při otevřeném oknu, kdy úspěšnost rozpoznávací řeči nebyla zvýšena jen u dvou uvedených nahrávek.

Tab. 24: Úspěšnost rozpoznávací řeči pro LMS a LMS + ICA (zavřená okna).

Nahrávky	Úspěšnost LMS [%]			Úspěšnost LMS +ICA [%]		
	80 km/h	100 km/h	130 km/h	80 km/h	100 km/h	130 km/h
Accept call	99	98	68	100	100	93
Decline call	99	100	100	100	100	99
Auto lights	100	98	92	100	99	98
Cruise control one hundred	100	99	92	100	98	99
Cruise control one hundred thirty	88	100	100	100	100	100
I stop on	97	89	26	100	100	100
I stop off	51	41	89	24	55	19
Radio on	78	81	75	99	100	99
Radio off	24	83	82	22	25	18

Tab. 25: Úspěšnost rozpoznávací řeči pro RLS a RLS + ICA (zavřená okna).

Nahrávky	Úspěšnost RLS [%]			Úspěšnost RLS +ICA [%]		
	80 km/h	100 km/h	130 km/h	80 km/h	100 km/h	130 km/h
Accept call	99	98	78	100	100	98
Decline call	100	99	98	100	100	99
Auto lights	100	99	88	100	99	99
Cruise control one hundred	99	99	98	100	100	99
Cruise control one hundred thirty	100	100	99	100	100	100
I stop on	98	23	29	100	98	97
I stop off	99	97	82	26	29	35
Radio on	88	98	89	99	99	92
Radio off	97	98	92	19	22	22

Tab. 26: Úspěšnost rozpoznávače řeči pro LMS a LMS + ICA (otevřené okno).

Nahrávky	Úspěšnost LMS [%]			Úspěšnost LMS +ICA [%]		
	80 km/h	100 km/h	130 km/h	80 km/h	100 km/h	130 km/h
Accept call	100	98	98	100	100	100
Decline call	100	100	99	100	100	100
Auto lights	100	99	99	100	99	99
Cruise control one hundred	100	100	98	100	100	99
Cruise control one hundred thirty	100	92	99	100	86	100
I stop on	100	99	99	100	100	100
I stop off	15	15	8	15	6	12
Radio on	99	100	98	99	99	100
Radio off	9	4	7	11	2	6

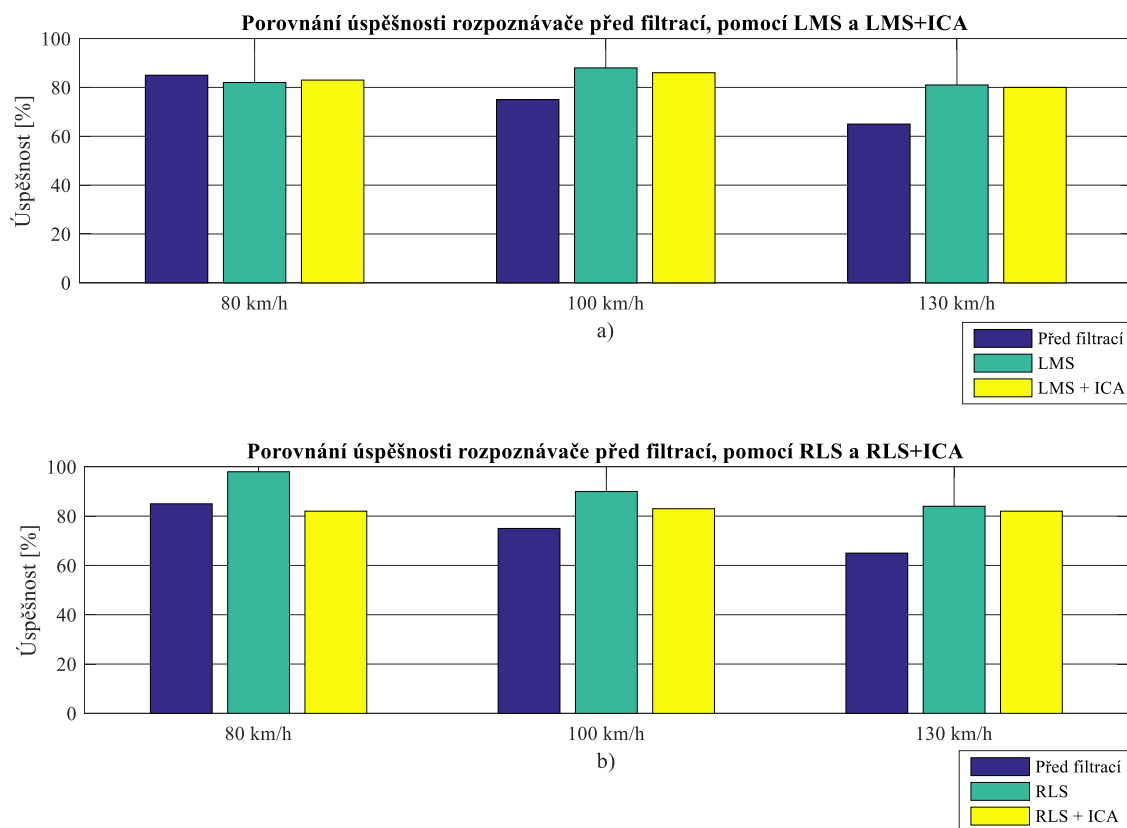
Tab. 27: Úspěšnost rozpoznávače řeči pro RLS a RLS + ICA (otevřené okno).

Nahrávky	Úspěšnost RLS [%]			Úspěšnost RLS +ICA [%]		
	80 km/h	100 km/h	130 km/h	80 km/h	100 km/h	130 km/h
Accept call	100	100	98	100	99	99
Decline call	100	100	99	100	100	100
Auto lights	100	99	99	100	100	99
Cruise control one hundred	100	100	100	100	100	99
Cruise control one hundred thirty	100	100	100	99	99	100
I stop on	55	68	65	98	56	72
I stop off	64	8	5	9	7	78
Radio on	88	98	85	100	99	90
Radio off	51	7	4	8	68	77

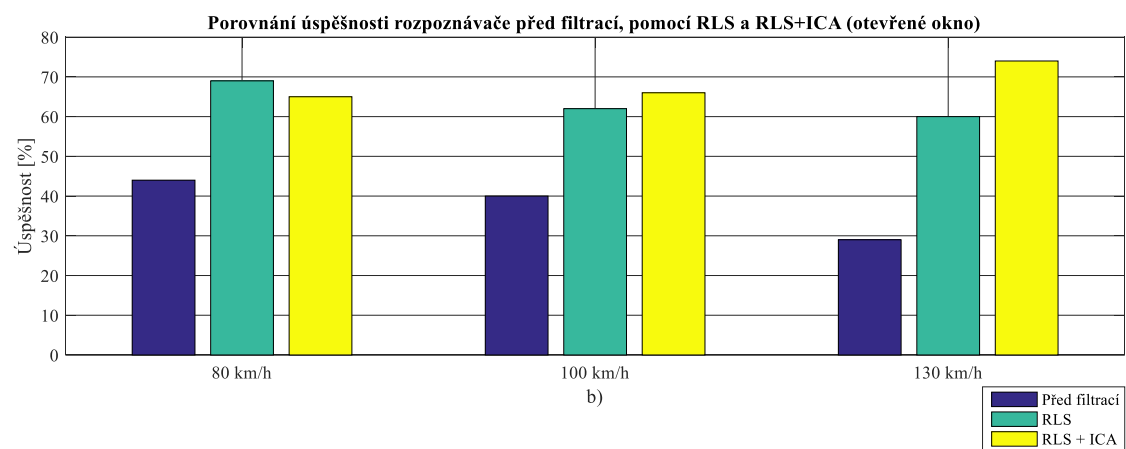
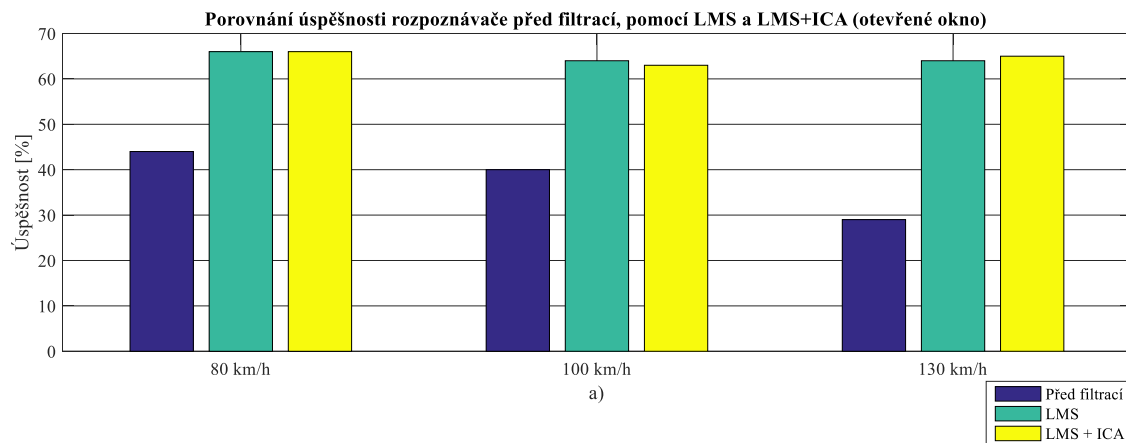
V grafech na Obr. 52 je zobrazena úspěšnost rozpoznávače řeči pro metody LMS a RLS. Při použití LMS algoritmu byla výsledná úspěšnost vždy vyšší než 80 %. Rozdílné rychlosti neměli téměř žádný vliv na úspěšnost při použití metody LMS. Kombinace LMS a ICA je zobrazena v grafu a). Výsledná úspěšnost byla vždy menší než při použití samotné metody a docházelo tedy spíše ke zhoršení úspěšnosti rozpoznávače. Na Obr. 52 b) je zobrazena kombinace metody RLS a ICA. Samotná metoda funguje velice dobře a lze pozorovat, že se zvyšující se rychlostí jedoucího automobilu výsledná úspěšnost rozpoznávače klesá. Při kombinaci této metody s metodou ICA je výsledná úspěšnost vždy menší a pohybuje se kolem 80 %. V grafech na Obr. 53 a) je zobrazena úspěšnost rozpoznávače při otevřených oknech pro LMS algoritmus. Výsledná úspěšnost se zvyšující se rychlostí zůstává stejná. To stejné platí i pro kombinaci adaptivní filtrace a metody ICA. Na Obr. 53 b) je zobrazena úspěšnost při použití RLS algoritmu. Pomocí tohoto algoritmu docházelo ke snížení výsledné úspěšnosti při porovnání s LMS. V kombinaci s metodou ICA bylo ovšem dosaženo již vyšších úspěšností rozpoznávání než při použití samotné adaptivní filtrace. Při



zavřených oknech je zde zobrazeno, že výsledná úspěšnost je lepší, při použití samostatné adaptivní filtrace. Kombinace s metodou nezávislých komponent výslednou úspěšnost vždy zhorší. Při jednom otevřeném oknu se již hodí použít tuto kombinaci, jelikož je výsledná úspěšnost rozpoznání vždy vyšší.



Obr. 52: Grafické zobrazení úspěšnosti rozpoznávače pro: a) kombinaci LMS a ICA, b) kombinaci RLS a ICA (zavřená okna).



Obr. 53: Grafické zobrazení úspěšnosti rozpoznávače pro: a) kombinaci LMS a ICA, b) kombinaci RLS a ICA (otevřené okno).

#### 4.7.5 Použití ve výrobní lince

Dalším místem pro posuzování kvality filtrace byla výrobní linka na výrobu autosedaček. Celkem bylo rozeznáváno 7 příkazů. Tyto příkazy jsou spíše ilustrační, a nejedná se o pravé příkazy, které jsou běžně užívány. Zde jsou zobrazeny použité příkazy:

- Homing.
- Turn on the laser welder.
- Turn off the laser welder.
- Turn on the main arm.
- Turn off the main arm.
- Turn on the press machine.
- Turn of the press machine.

Ve výrobní lince se nacházeli dva hlavní zdroje hluku. Prvním zdrojem hluku byla laserová svářečka a jsou zde použity příkazy pro ovládání tohoto stroje. Dalším druhem hluku byl velký lis, který sloužil k vytlačování a tvarování oceli a opět jsou použity příkazy pro ovládání tohoto lisu. Poslední příkazy slouží k ovládání posuvného ramena.

Referenční nahrávky byly vytvořeny v bezdovukové komoře, pro absolutní čistotu bez hluku okolí. Při použití těchto nahrávek s daným hlukem, tedy při spojení nahrávky řeči a hluku, nebyla řeč téměř vůbec slyšitelná a filtrace neměla žádný úspěch. Z tohoto důvodu byly nahrávky řeči programově zesíleny, aby bylo dosaženo zvýšení úrovně řeči v silně hlučném prostředí. Byla tím pouze simulována situace, když bychom se pokoušeli mluvit v takovémto prostředí v klasické řečové úrovni, posluchač by nám vůbec nerozuměl. Proto došlo k zesílení řeči při kontaminaci s hlukem svářečky na trojnásobek původní úrovně a pro kontaminaci s hlukem lisu dokonce na desetinásobek původní úrovně řeči. Při originální nahrávce řeči v prostředí lisu, byla úroveň řeči v takových hodnotách, které by při naší simulaci odpovídali čtrnáctinásobku originální nahrávky. Pro posuzování kvality řeči byla ovšem vybrána nižší úroveň řeči, aby bylo možné případně řeč ještě zvýšit.

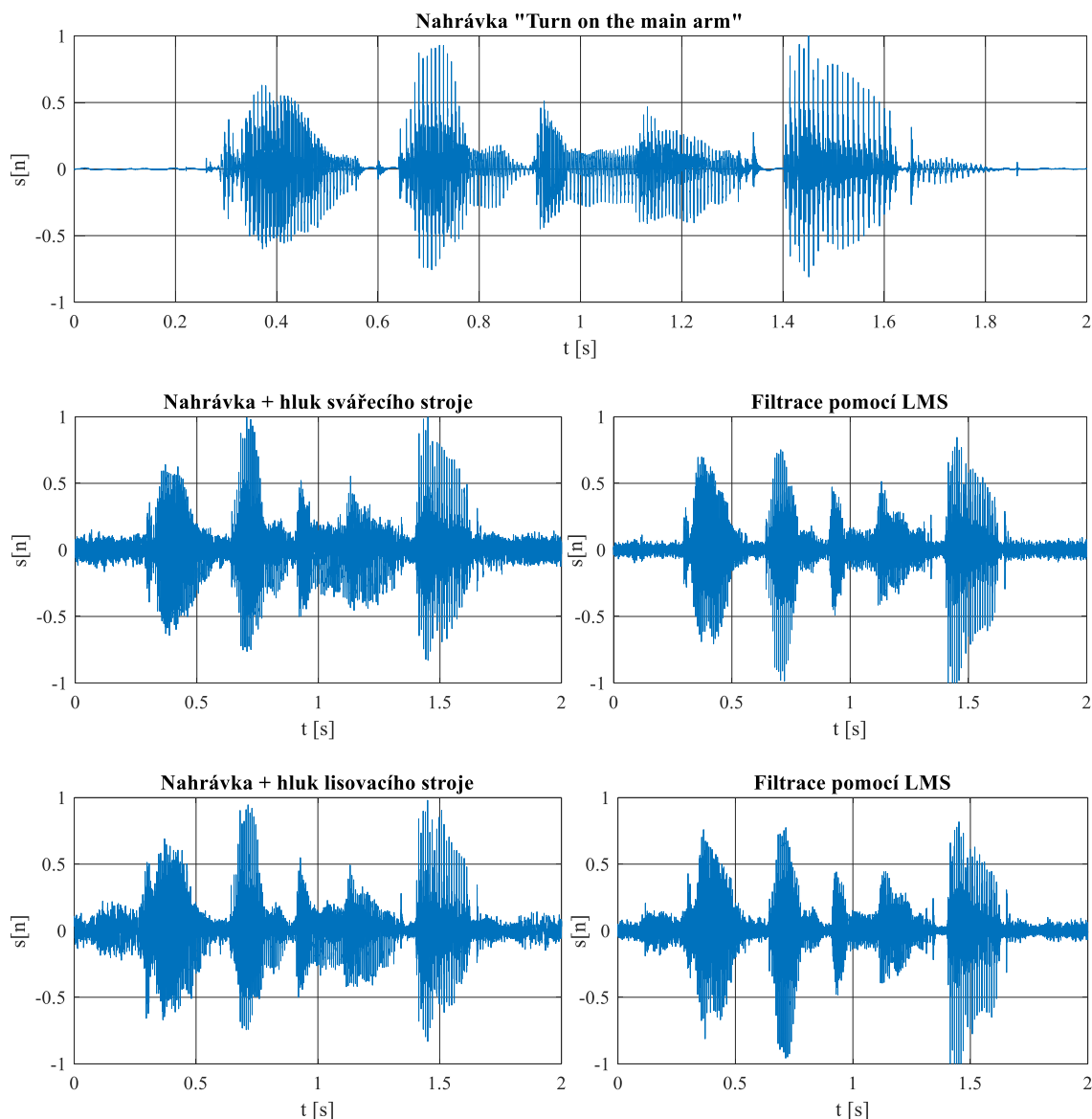
V Tab. 28 a Tab. 29 jsou uvedeny výsledky pro filtraci pomocí LMS algoritmu. Tabulky jsou rozděleny na tři části. První část zobrazuje hodnoty řádu filtru, které jsou nastaveny jako ideální z pohledu hodnot  $SNR$  po filtraci. Druhá část zobrazuje hodnoty  $SNR$ ,  $SSNR$  a úspěšnost rozpoznávací řeči WSR před filtrací. Ve třetí části jsou zobrazeny hodnoty  $SNR$ ,  $SSNR$  a úspěšnost WSR po provedené filtraci. Průměrné hodnoty  $SNR$  před filtrací pro hluk svářecího stroje jsou 5,31 dB a  $SSNR$  je 7,17 dB. Pomocí filtrace LMS algoritmem, byla hodnota  $SNR$  zvýšena na 12,22 dB a hodnota  $SSNR$  byla zvýšena na 14,15 dB. Oproti hodnotám před filtrací lze pozorovat zlepšení o téměř 7 dB pro obě uvedené hodnoty. Pro hluk lisovacího stroje je průměrná hodnota  $SNR$  3,98 dB a  $SSNR$  je 11,41 dB. Pomocí filtrací byla hodnota  $SNR$  zvýšena na 10 dB a  $SSNR$  na 16,56 dB. Hodnoty po filtraci byly zlepšeny o 6 dB pro  $SNR$  a o 5 dB pro hodnoty  $SSNR$ . Průměrná hodnota úspěšnosti rozpoznávací řeči byla před filtrací 42 % pro hluk svářecího stroje a pro hluk lisovacího stroje byla tato hodnota 57 %. Co se týká hodnot po filtraci, lze pozorovat zlepšení na 88 % pro příkazy u svářecího stroje. Pro lisovací stroj byla úspěšnost po filtraci zlepšena na 89 %.

Tab. 28: Výsledky filtrace a úspěšnost rozpoznávače pro hluk svářečského stroje (LMS).

Filtrace LMS	Výsledky pro hluk svářečského stroje, $\mu = 0,001$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Homing	90	3,85	7,83	0	9,52	12,77	55
Turn on the laser welder	40	6,22	7,25	0	13,31	14,32	75
Turn off the laser welder	40	5,39	6,70	8	12,57	13,60	95
Turn on the main arm	80	5,83	7,59	88	12,89	15,23	98
Turn off the main arm	40	3,39	5,14	78	10,45	12,68	97
Turn on the press machine	110	6,44	8,08	54	13,65	15,46	100
Turn off the press machine	40	6,07	7,62	66	13,17	14,97	98

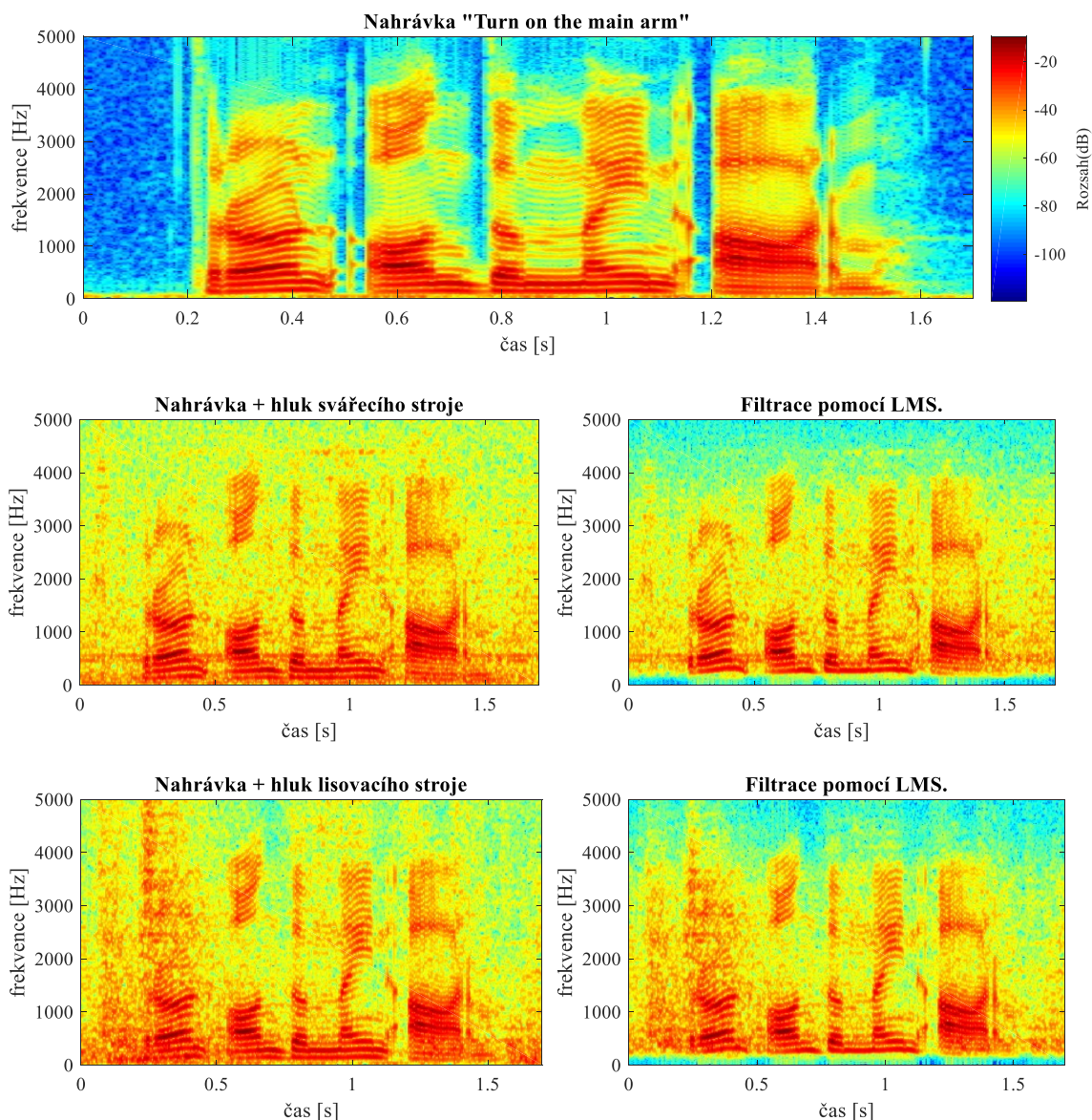
Tab. 29: Výsledky filtrace a úspěšnost rozpoznávače pro hluk lisovacího stroje (LMS).

Filtrace LMS	Výsledky pro hluk lisovacího stroje, $\mu = 0,001$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Homing	50	1,02	10,39	0	6,17	13,85	42
Turn on the laser welder	20	5,87	12,93	5	11,81	18,07	91
Turn off the laser welder	50	5,41	13,99	84	11,50	19,92	95
Turn on the main arm	40	5,77	10,64	62	11,84	15,86	98
Turn off the main arm	20	5,20	10,89	77	11,19	15,42	97
Turn on the press machine	30	4,03	12,54	81	10,23	18,12	99
Turn off the press machine	50	0,57	8,48	89	7,17	14,69	98



Obr. 54: Časové průběhy před a po filtraci za použití LMS algoritmu, v továrně.

Na Obr. 54 jsou zobrazeny výsledky filtrace pomocí LMS algoritmu při použití v továrně. V grafech se nachází nahrávka „*Turn on the main arm*“, pro hluk svářecího a lisovacího stroje. V levé části se nachází nahrávky, které jsou kontaminovány daným hlukem. Pravá část zobrazuje časové průběhy po filtraci za použití LMS algoritmu. Obr. 55 zobrazuje výsledky filtrace v časově-frekvenční oblasti. Výsledky zobrazují, že docházelo ke zlepšení nahrávek po filtraci. U nahrávky kontaminované hlukem lisovacího stroje, se na začátku nachází právě bouchnutí tohoto stroje. Tato úroveň hluku se nachází i v nahrávce po filtraci. Filtrace si tedy nedokázala zcela dobře poradit s takto silným rušením.



Obr. 55: Testované signály zobrazené v časově-frekvenční oblasti před a po filtraci pomocí LMS algoritmu znázorněné ve spektrogramu.

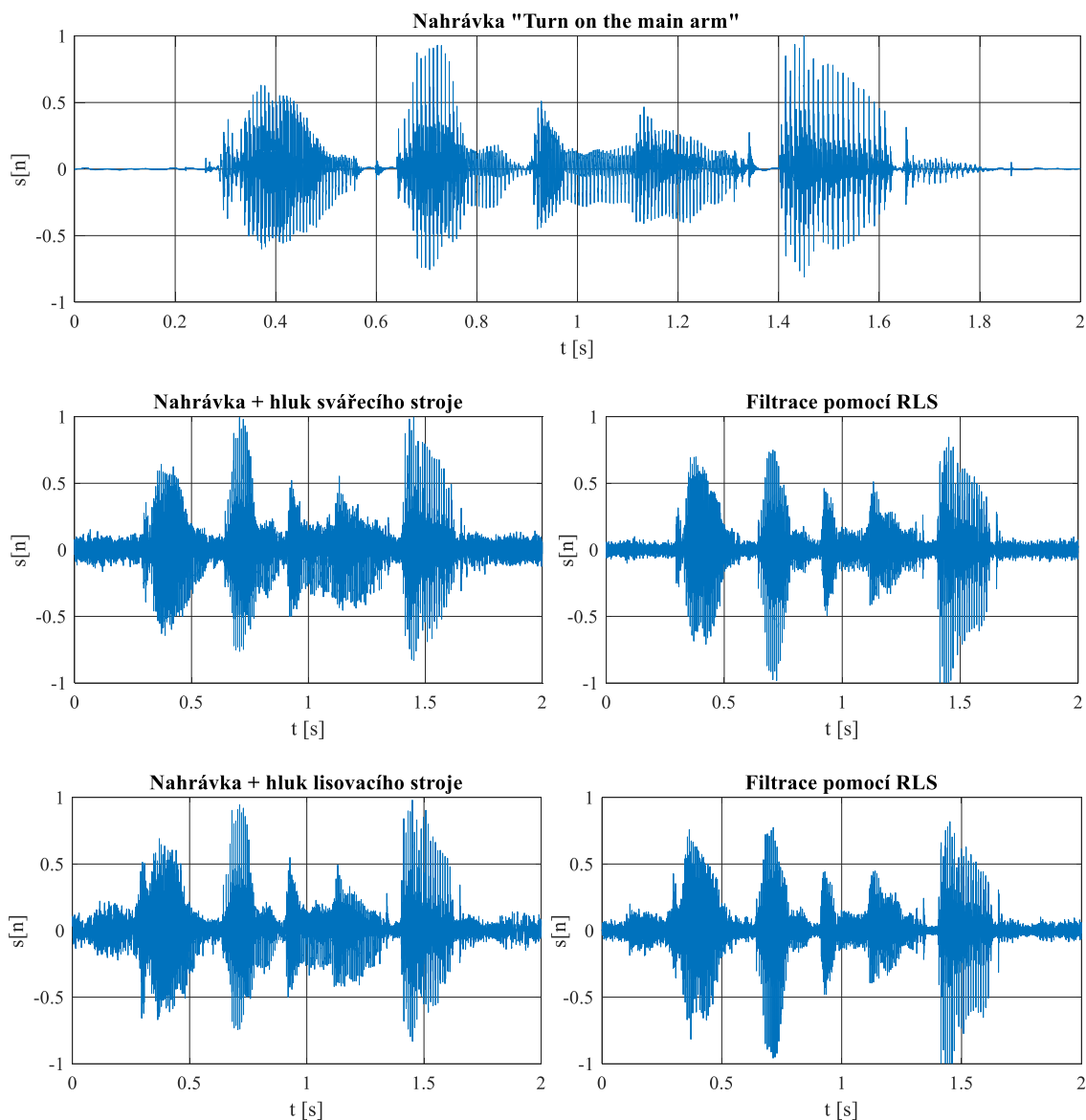
Následně byla provedena filtrace za použití RLS algoritmu. Hodnota faktoru zapomínání byla stanovena jako  $\lambda = 1$ . Hodnoty řádů filtru se pohybují v rozsahu od 10 do 70. Z časového hlediska to představuje doby výpočtu do 3 sekund. V Tab. 30 a Tab. 31 Jsou uvedeny výsledky pro filtraci pomocí RLS algoritmu. Průměrné zlepšení hodnoty  $SNR$  bylo z 5,3 dB na 12,3 dB pro hluk svářecího stroje. Pro hluk lisovacího stroje bylo proběhlo zlepšení  $SNR$  z 4 dB na 10 dB. Co se týče hodnot  $SSNR$  bylo zlepšení z hodnoty 7,1 dB na 14,1 dB pro svářecí stroj a pro lisovací stroj proběhlo zlepšení z 11,4 dB na 16,75 dB. Pro hodnoty úspěšnosti rozpoznávací řeči lze pozorovat zlepšení ze 42 % na 94 % u hluku svářecího stroje. Pro hluk lisovacího stroje byla hodnota před filtrací rovna 57 % a po filtraci pomocí RLS došlo ke zlepšení na 84 %.

Tab. 30: Výsledky filtrace a úspěšnost rozpoznávače pro hluk svářecího stroje (RLS).

Filtrace RLS	Výsledky pro hluk svářecího stroje, $\lambda = 1$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Homing	50	3,85	7,83	0	9,31	12,53	69
Turn on the laser welder	50	6,22	7,25	0	13,32	14,85	91
Turn off the laser welder	60	5,39	6,70	8	12,65	13,94	99
Turn on the main arm	70	5,83	7,59	88	13,67	14,9	100
Turn off the main arm	60	3,39	5,14	78	10,56	12,79	99
Turn on the press machine	50	6,44	8,08	54	13,63	15,49	99
Turn off the press machine	30	6,07	7,62	66	13,07	14,69	98

Tab. 31: Výsledky filtrace a úspěšnost rozpoznávače pro hluk lisovacího stroje (RLS).

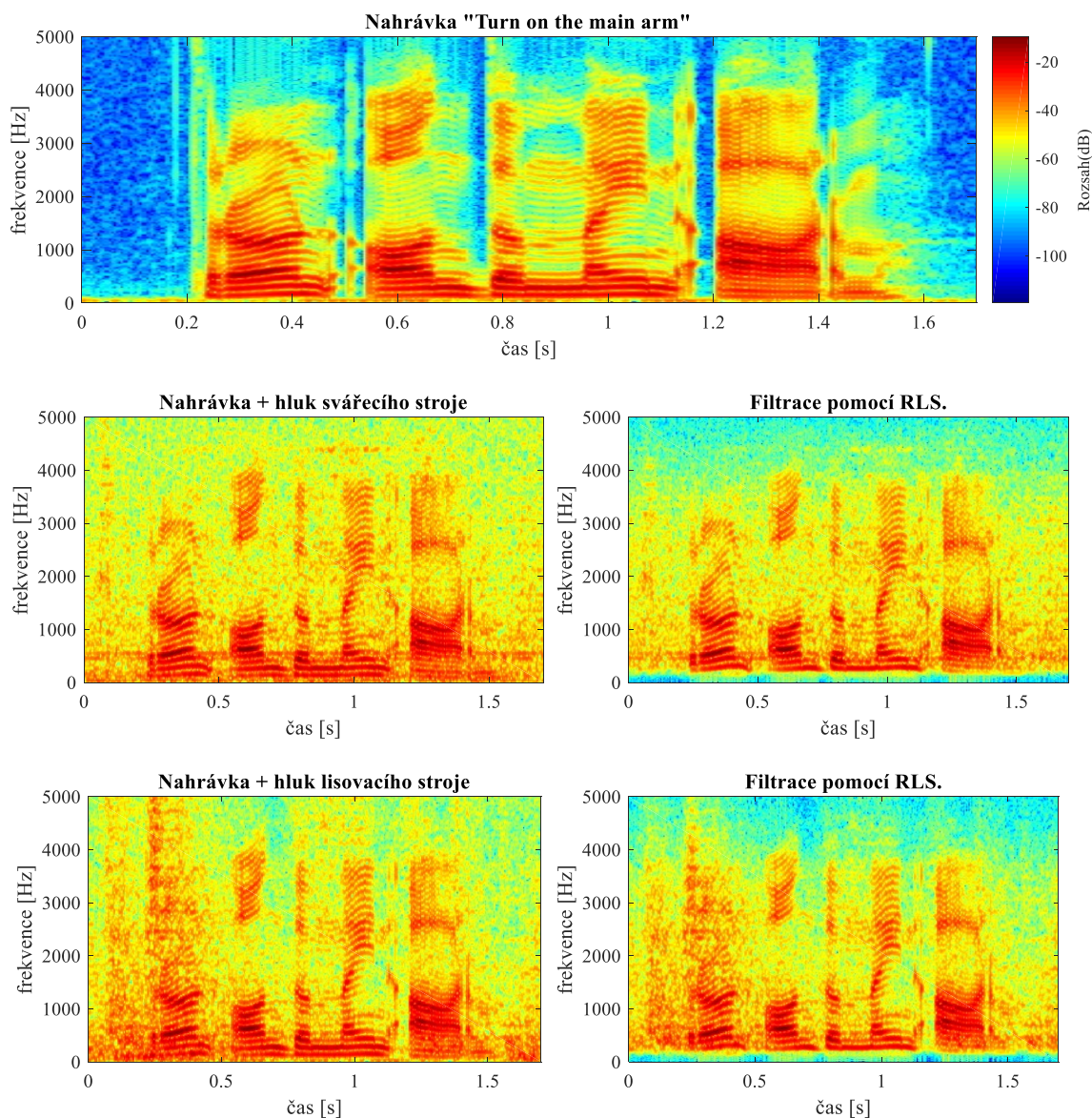
Filtrace RLS	Výsledky pro hluk lisovacího stroje, $\lambda = 1$						
	Řád filtru	Před filtrací			Po filtraci		
		SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]	SNR [dB]	SSNR [dB]	Úspěšnost WSR [%]
Homing	50	1,02	10,39	0	6,16	14,55	10
Turn on the laser welder	60	5,87	12,93	5	11,76	18,59	98
Turn off the laser welder	30	5,41	13,99	84	11,46	19,94	96
Turn on the main arm	40	5,77	10,64	62	11,84	15,92	98
Turn off the main arm	20	5,2	10,89	77	11,09	15,4	99
Turn on the press machine	10	4,03	12,54	81	10,21	18,11	98
Turn off the press machine	20	0,57	8,48	89	7,17	14,72	95



Obr. 56: Časové průběhy před a po filtraci za použití RLS algoritmu, v továrně.

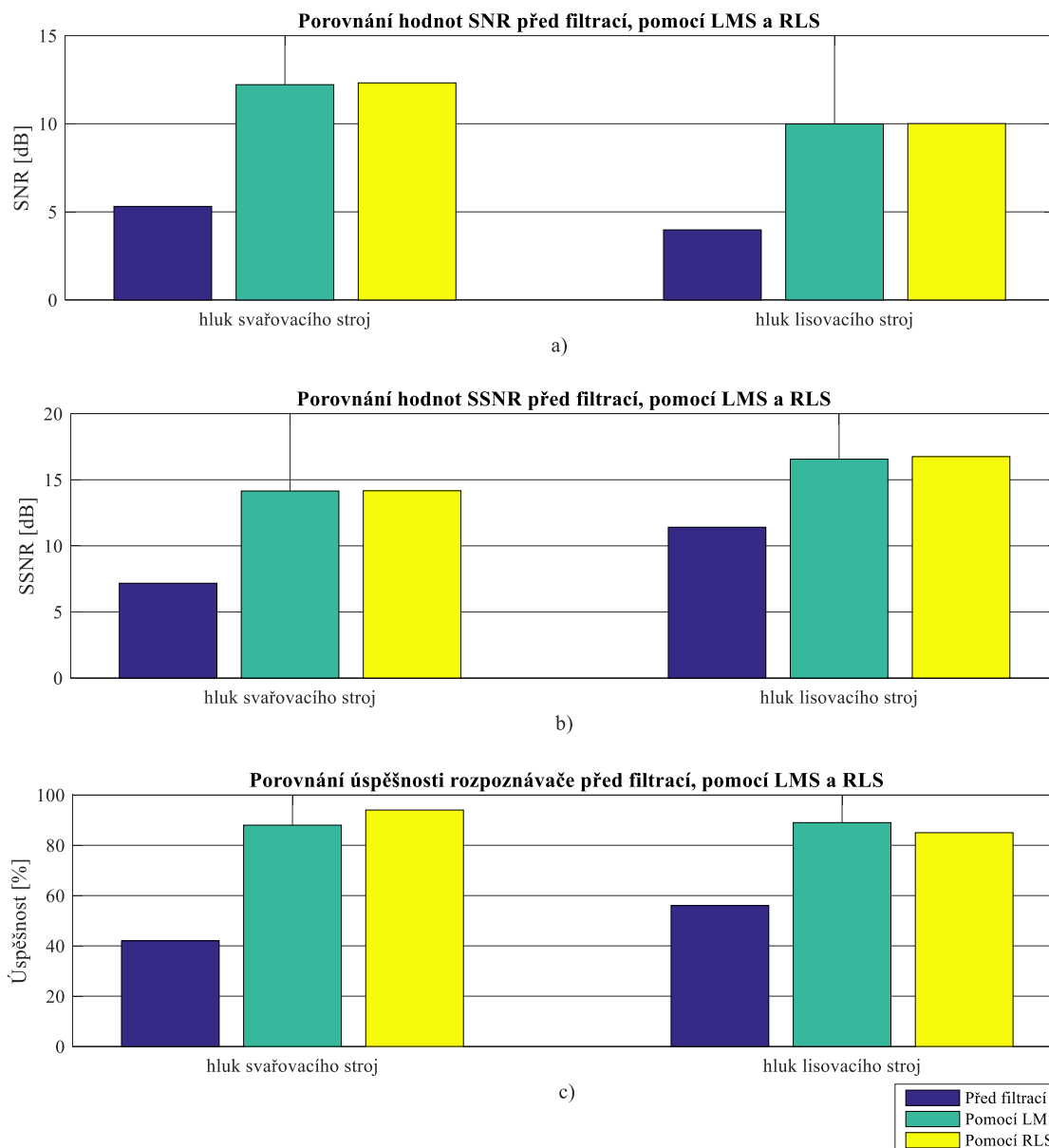
Obr. 56 zobrazuje výsledky filtrace pomocí RLS algoritmu při použití v továrně. V grafech se nachází nahrávka „*Turn on the main arm*“, pro hluk svářečského a lisovacího stroje. V levé části se nachází nahrávky, které jsou kontaminovány daným hlukem. Pravá část zobrazuje časové průběhy po filtraci za použití RLS algoritmu. Obr. 57 zobrazuje výsledky filtrace v časově-frekvenční oblasti. Stejně jako při použití LMS algoritmu tak i zde lze v grafech pozorovat, že u nahrávky kontaminované hlukem lisovacího stroje, nebyla správně odfiltrován hluk bouchnutí lisu.





Obr. 57: Testované signály zobrazené v časově-frekvenční oblasti před a po filtraci pomocí RLS algoritmu znázorněné ve spektrogramu.

Následující grafy (Obr. 58) zobrazují porovnání výsledků filtrace pro oba použité algoritmy. Graf a) zobrazuje průměrné hodnoty  $SNR$  všech použitých nahrávek před filtrací, za použití LMS a RLS algoritmu. V grafu b) jsou zobrazeny průměrné hodnoty  $SSNR$  a třetí graf c) zobrazuje úspěšnost rozpoznávací řeči. V levé části se nachází výsledky pro hluk svařovacího stroje a v pravé části jsou zobrazeny výsledky pro hluk lisovacího stroje. Při použití adaptivní filtrace došlo k výraznému zlepšení hodnot  $SNR$ ,  $SSNR$  tak i úspěšnosti rozpoznávací řeči. V tomto případě nelze pozorovat rozdíl mezi algoritmem LMS a RLS a filtrace dosahuje stejných výsledků.



Obr. 58: Porovnání výsledků filtrace pro: a) SNR, b) SSNR, c) Úspěšnost WSR, pro algoritmy LMS a RLS, použití v továrně.

#### 4.7.6 Kombinace adaptivní filtrace a ICA

V následující kapitole se nachází výsledky při použití kombinace adaptivní filtrace a metody nezávislých komponent. V Tab. 32, Tab. 33 je zobrazena úspěšnost rozpoznávače řeči. Tabulky jsou rozděleny na dvě části. První část zobrazuje výsledky, kdy byla použita pouze adaptivní filtrace. Druhá část zobrazuje právě danou kombinaci s metodou ICA. V tabulkách lze pozorovat, že úspěšnost rozpoznávače řeči se pohybovala nad 90 %. Výsledky při použití LMS a RLS algoritmu jsou téměř shodné. Jediná nahrávka, u níž nedošlo ke zlepšení, byla „Homing“ při hluku lisovacího stroje. Při poslechové metodě, byla nahrávka zřetelně slyšitelná, ale jelikož se v pozadí

nacházela stále úroveň hluku bouchání lisovacího stroje, rozpoznávače nedokázal správně určit danou nahrávku.

Tab. 32: Úspěšnost rozpoznávače řeči pro LMS a LMS + ICA.

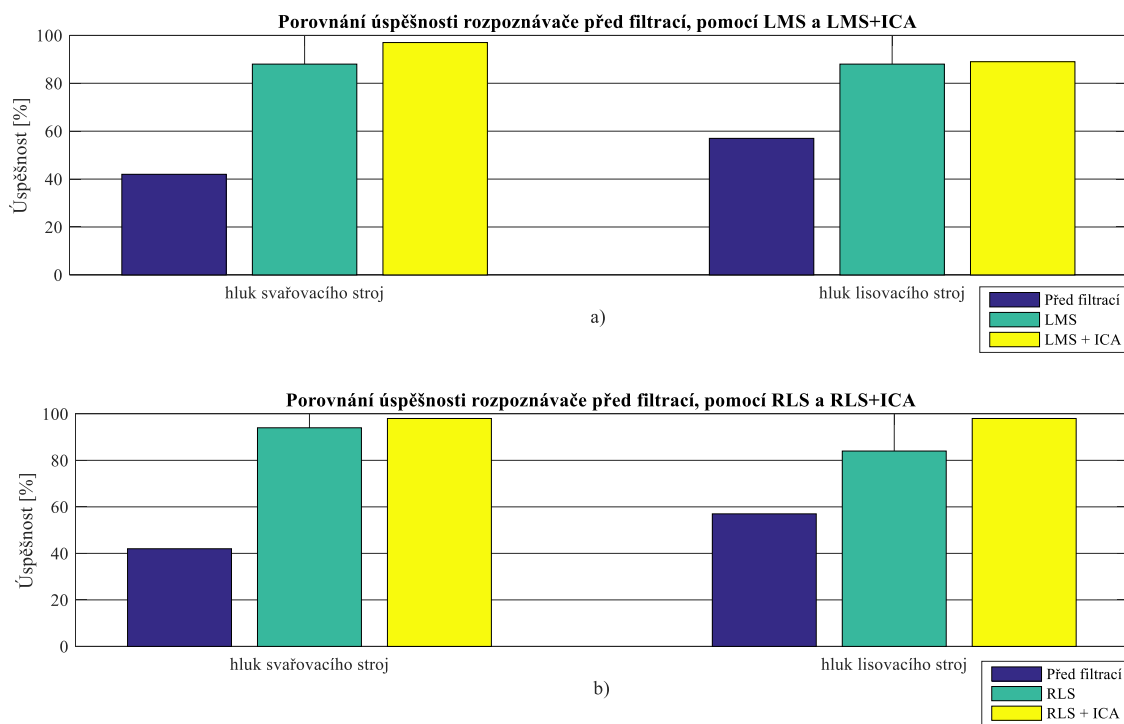
Nahrávky	Úspěšnost LMS [%]		Úspěšnost LMS + ICA [%]	
	Hluk svářecího stroje	Hluk lisovacího stroje	Hluk svářecího stroje	Hluk lisovacího stroje
Homing	55	42	98	35
Turn on the laser welder	75	91	98	99
Turn off the laser welder	95	95	91	100
Turn on the main arm	98	98	99	99
Turn off the main arm	97	97	96	98
Turn on the press machine	100	99	100	99
Turn off the press machine	98	98	97	98

Tab. 33: Úspěšnost rozpoznávače řeči pro RLS a RLS + ICA.

Nahrávky	Úspěšnost RLS [%]		Úspěšnost RLS + ICA [%]	
	Hluk svářecího stroje	Hluk lisovacího stroje	Hluk svářecího stroje	Hluk lisovacího stroje
Homing	69	10	98	99
Turn on the laser welder	91	98	99	99
Turn off the laser welder	99	96	98	98
Turn on the main arm	100	98	100	99
Turn off the main arm	99	99	99	98
Turn on the press machine	99	98	99	99
Turn off the press machine	98	95	99	99

Na Obr. 59 je graficky znázorněna úspěšnost rozpoznávače řeči pro metody LMS a RLS. Při použití LMS algoritmu byla výsledná úspěšnost 88 % pro oba použité hluky. Při kombinaci s metodou ICA lze vidět zlepšení na 97 % u hluku svářecího stroje a 89 % pro lisovací stroj. Tato kombinace je zobrazena na Obr. 59 a). Na Obr. 59 b) je zobrazena kombinace metody RLS a ICA. Samotná metoda RLS měla vliv na zlepšení úspěšnosti rozpoznávače na 94 % u svářecího stroje a 85 % u

lisovacího stroje. Při kombinaci s metodou ICA docházelo ke zlepšení na 98 % pro oba použité hluky továrny. Na rozdíl od použití v automobilu, zde docházelo ke zlepšení při použití kombinace ICA a dané adaptivní filtrace.



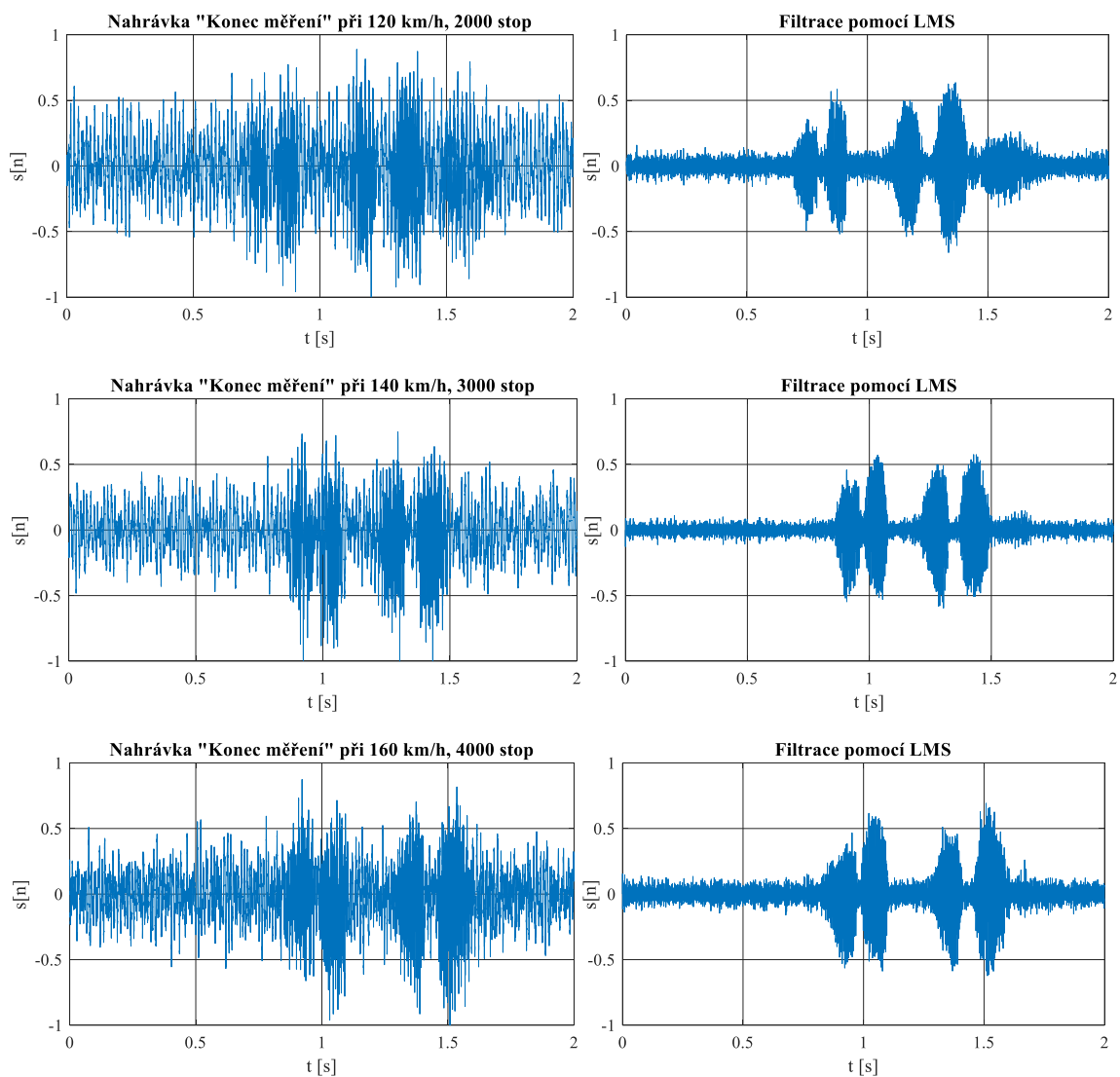
Obr. 59: Grafické zobrazení úspěšnosti rozpoznávače pro: a) kombinaci LMS a ICA, b) kombinaci RLS a ICA.

#### 4.7.7 Použití v letadle

Při posuzování kvality filtrace v letadle nebyly předem určeny a vytvořeny referenční nahrávky. Ve výsledcích se tedy budou vyskytovat příkazy, které nebyly nahrány v částečně bezdovukové komoře, ale přímo v kokpitu letadla. V hlasových nahrávkách byl vždy uveden příkaz „*Začátek měření*“, dále byly ohlášeny informace o stavu výšky a rychlosti letadla. Nahrávání bylo ukončeno příkazem „*konec měření*“.

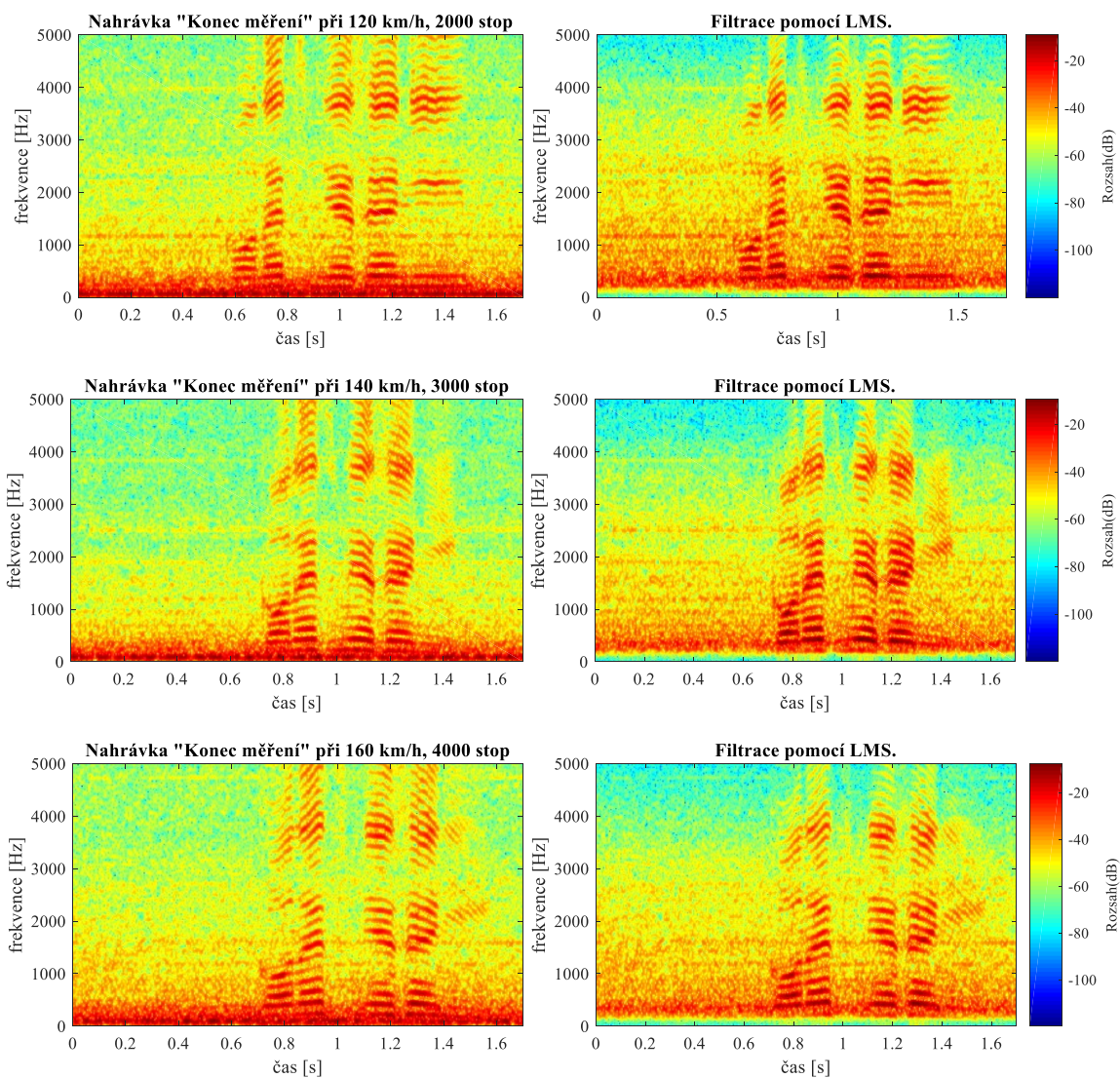
Dané nahrávky byly provedeny celkem při třech různých rychlostech. Jednalo se o rychlosti 120 km/h, 140 km/h a 160 km/h. Pro každou rychlost proběhlo měření v nadmořské výšce 2000 stop, 3000 stop a 4000 stop. Vzhledem k tomu, že nebyly předem nahrány referenční nahrávky řeči, nebudou zde uvedeny výsledky odstupů signálu od šumu. Výsledky budou poukazovat na úspěšnost filtrace pomocí poslechových metod.

Na Obr. 60 se nachází časové průběhy výsledků filtrace za použití LMS algoritmu. V levé části jsou umístěny průběhy v rychlostech 120 km/h, 140 km/h a 160 km/h s postupným zvedáním letové výšky. Podle chybějících referenčních nahrávek řeči, se nedají určit hodnoty *SNR*. Podle poslechových metod, byly slyšitelné rozdíly. Tato metoda filtrace si dokázala částečně poradit s danými úrovněmi hluku v letadle. Zvolený rozpoznávač řeči WSR je schopen rozpoznávat jen určité jazyky, které jsou uvedeny v kapitole 4.4.2. Z toho důvodu nebylo možné použít nahrávky, které byly nahrány v českém jazyce pro posouzení úspěšnosti rozpoznávače řeči. Nebylo by totiž dokázáno, zda rozpoznávač má zhoršenou úspěšnost z důvodu špatné filtrace nebo zda je to způsobeno jeho malou slovní zásobou. Obr. 61 zobrazuje dané výsledky filtrace v časově-frekvenční oblasti. V těchto grafech lze pozorovat stále silnou úroveň hluku v pozadí. Podle výsledků filtrace zobrazených v grafech lze říct, že úspěšnost nebyla příliš vyhovující. Z hlediska poslechových metod docházelo k rozdílům před a po filtraci. Tyto rozdíly byly znatelné a dalo by se říct, že pokud by bylo možné vyhodnotit dané příkazy pomocí rozpoznávače řeči, docházelo by ke zlepšení.



Obr. 60: Časové průběhy před a po filtraci za použití LMS algoritmu, v letadle.





Obr. 61: Testované signály zobrazené v časově-frekvenční oblasti před a po filtraci pomocí LMS algoritmu znázorněné ve spektrogramu.

## Závěr

Prvotním cílem diplomové práce bylo zpracování rešerše a na jejím základě zvolit vhodné metody. Jako metody, které byly pomocí této rešerše vybrány, byla metoda adaptivní filtrace a metoda nezávislých komponent ICA. U adaptivní filtrace byly použity algoritmy LMS a RLS. Jedná se o vícekanálové metody, tudíž při zpracovávání bylo třeba mít k dispozici více vstupních nahrávek. Jednotlivé metody jsou popsány v teoretické části. Během realizace bylo třeba nejdříve vytvořit aplikaci, která slouží k nahrávání jednotlivých nahrávek řeči a šumu. Poté bylo zpracováno uživatelské rozhraní druhé aplikace, která slouží k zpracovávání těchto nahrávek. Obě aplikace byly vytvořeny v programu LabVIEW. Rozhraní GUI slouží k vizualizaci použitých nahrávek řeči a šumu, a jejich vzájemné kontaminaci. Dále jsou zde zobrazeny výsledky zvolené filtrace jak v časové, tak v časově-frekvenční oblasti. Výsledky filtrace jsou vyhodnoceny pomocí hodnot odstupe signálu od šumu  $SNR$  i  $SSNR$ . Poslední vizualizací výsledků je úspěšnost rozpoznávače řeči WSR. Popis funkčnosti daného programu se nachází v experimentální části diplomové práce.

Při vyhodnocování výsledků rozpoznávače řeči z hlediska jeho úspěšnosti, bylo třeba nejdříve zjistit jeho funkčnost. Jelikož se nedá zjistit jeho přesná vnitřní struktura a chová se tedy spíše jako „black-box“, bylo třeba uskutečnit sérii testů pro zajištění jeho nejlepší funkčnosti. Po uskutečnění těchto testů bylo zjištěno, že je třeba použít předfiltraci pomocí Butterworth filtru, kdy pomocí této filtrace byla zlepšena úspěšnost rozpoznávání na 98 %. Při druhém testu byla zjištěna hodnota amplitudy daných nahrávek, kdy v ideálním případě by měla být v rozsahu od -1 do 1.

Po správném nastavení rozpoznávače řeči, byly zpracovány jednotlivé výsledky. Měření nahrávek probíhalo v celkem čtyřech různých oblastech, ve kterých bylo zkoumáno použití hlasového ovládání. Konkrétně se jednalo o použití v osobním automobilu, v domácnosti, v továrně, a nakonec i v letadle. Pro každou oblast bylo vybráno několik příkazů, pomocí kterých byly vyhodnocovány výsledky filtrace podle hodnot  $SNR$  a úspěšnost rozpoznávače řeči. Každá nahrávka byla testována celkem stokrát.

V jednotlivých oblastech, bylo při vyhodnocování filtrace použito dvou adaptivních algoritmů. Jednalo se o algoritmy LMS a RLS, kdy první jmenovaný se vyznačuje zejména svojí jednoduchostí a matematickou nenáročností. Druhý algoritmus RLS se vyznačuje rychlou adaptivitou na signál, za kterou platí svojí matematickou náročností. Při použití těchto algoritmů pro filtraci v domácnosti, vykazoval RLS algoritmus vždy lepších hodnot  $SNR$  a  $SSNR$  než LMS. Úspěšnost rozpoznávače řeči byla v průměru 60 % pro LMS algoritmus a 90 % pro RLS. Je tedy patrné, že RLS algoritmus si dokázal lépe poradit s filtrací hluků pračky, ventilátoru a myčky. Následně byla použita kombinace adaptivní filtrace AF a metody nezávislých komponent ICA. Kombinací algoritmu LMS a ICA byla dosažena výsledná úspěšnost rozpoznávače řeči 96 %. Při použití kombinace s RLS algoritmem byla výsledná úspěšnost vyšší než 99 % pro jednotlivé nahrávky.

Další zkoumanou oblastí byl osobní automobil, kde byla zkoumána úspěšnost filtrace se zvyšující se rychlostí vozidla. Zde již nebylo vidět výrazných rozdílů mezi LMS a RLS algoritmem ve výsledcích filtrace pomocí hodnot  $SNR$ . U segmentální hodnoty  $SNR$  bylo vidět zlepšení o více než 2 dB. Z hlediska úspěšnosti rozpoznávače řeči byla průměrná úroveň rozpoznání 83 % pro LMS a 91 % pro RLS algoritmus. V kombinaci AF a ICA docházelo při použití v automobilu vždy k nepatrnému zhoršení výsledků. Výsledná úspěšnost rozpoznávače řeči byla lepší při použití



samotné adaptivní filtrace v kombinaci s metodou ICA docházelo ke zhoršení rozpoznávání při zavřených oknech. Pro otevřené okno byla průměrná úspěšnost rozpoznávače 37 % před filtrací. Pomocí algoritmu LMS a RLS byla úspěšnost zvýšena na 64 %. Kombinací LMS a ICA zůstala úspěšnost rozpoznávače řeči stejná. Pro kombinaci RLS a ICA došlo ke zlepšení úspěšnosti rozpoznávače řeči na 68 %. Na rozdíl od zavřených oken, lze zde pozorovat zlepšení při použití kombinace AF a metody nezávislých komponent.

Následující oblastí byla továrna, ve které se vyskytovaly dva hlavní zdroje hluku. Jednalo se o hluk svářecího a lisovacího stroje. Při použití originálních nahrávek v továrně by nebylo možné provádět filtraci, jelikož úroveň hluku je zde příliš velká. Z toho důvodu byly nahrávky, které byly nahrány v běžné řečové úrovni zesíleny a tím byla provedena simulace zvýšení úrovně řeči na pracovišti. Při použití LMS algoritmu docházelo ke zlepšení *SNR* o hodnotu 11 dB a pro *SSNR* bylo zlepšení o 15,5 dB. Stejných výsledků bylo dosaženo i při použití RLS algoritmu. Úspěšnost rozpoznávače řeči byla pro LMS a RLS algoritmus 89 %. V kombinaci s metodou ICA bylo dosaženo zlepšení na hodnotu 93 % při kombinaci LMS a ICA a při kombinaci s RLS algoritmem byla výsledná úspěšnost 98 %. Na rozdíl od použití v automobilu, zde docházelo ke zlepšení při použití kombinace ICA a dané adaptivní filtrace.

Poslední zkoumanou oblastí bylo letadlo. Byl zde zkoumán vliv výšky a rychlosti letadla na výslednou filtraci pomocí LMS algoritmu. Vzhledem k tomu, že nahrávky určené pro letadlo nebyly předem vytvořeny v bezdovukové komoře, nebylo možné vypočítat hodnoty *SNR* z důvodu chybějící referenční nahrávky řeči. Podle poslechových metod, byly ovšem slyšitelné rozdíly. Tato metoda filtrace si dokázala částečně poradit s danými úrovněmi hluku v letadle. Zvolený rozpoznávač řeči WSR je schopen rozpoznávat jen určité jazyky. Z toho důvodu nebylo možné použít nahrávky, které byly nahrány v českém jazyce pro posouzení úspěšnosti rozpoznávače řeči. Nebylo by totiž dokázáno, zda rozpoznávač má zhoršenou úspěšnost z důvodu špatné filtrace nebo zda je to způsobeno jeho malou slovní zásobou.

Algoritmus RLS se tedy jeví jako lepší, jelikož dosahuje lepších výsledků při porovnání s LMS algoritmem. To je důvodem proč RLS metoda vykazuje výbornou kvalitu zvuku. Vzhledem k tomu, že se RLS algoritmus vyznačuje vysokou matematickou náročností oproti LMS, je použití tohoto algoritmu v reálném prostředí obtížné. Ve výsledcích lze pozorovat i úkaz, ke kterému docházelo při zpracovávání nahrávek končící slovem „off“. U nahrávek, které jsou dobře rozpoznatelné i před použitím filtrace, dochází při filtraci k deformaci dané nahrávky a energie slova „off“ se v průběhu filtrace ztratí.

## Seznam použité literatury

- [1] RABINER, L., SCHAFER, R. *Automatic speech recognition (ASR)*. Santa Barbara, Rutgers University and the University of California, 2004.
- [2] YU, D., DENG, L. *Automatic speech recognition*, ISBN: 978-1-4471-5779-3, London, Springer-Verlag, 2015.
- [3] RABINER, L., JUANG, B. H. *Fundamentals of Speech Recognition*. ISBN: 0-13285826-6, New Jersey, Saddle River, 1993.
- [4] WAIBEL, A., LEE, K. F. *Readings in Speech Recognition*, ISBN 1-55860-124-4 California, San Mateo, Morgan Kaufmann 1990.
- [5] CHOI, E. *Noise robust front-end for ASR using spectral subtraction, spectral flooring and cumulative distribution mapping*. Australian International Conference on Speech Science & Technology, Australia, Sydney, 2004.
- [6] HERMAN, H., SHARMA, S. *Temporal patterns (TRAPs) in ASR of noisy speech*. ICASSP'99, Phoenix, Arizona, USA, 1999. [online]. Dostupné z: <https://pdfs.semanticscholar.org/bce4/7e40ed085d1096ec633cad7593a8ac54329.pdf>
- [7] KIM, S., RAJ, B., LANE, I. *Environmental noise embeddings for robust speech recognition*. Carnegie Mellon University, 2016.
- [8] FUKANE, A. R., SAHARE, S. L., *Noise estimation Algorithm for Speech enhancement in highly non-stacionary Enviroments*. *International Journal of Computer Science Issues (IJCSI)*, 2011. [online]. Dostupné z: [https://www.researchgate.net/publication/260319703\\_Noise\\_estimation\\_Algorithms\\_for\\_Speech\\_Enhancement\\_in\\_highly\\_non-stationary\\_Environments](https://www.researchgate.net/publication/260319703_Noise_estimation_Algorithms_for_Speech_Enhancement_in_highly_non-stationary_Environments)
- [9] PSUTKA, J., MÜLLER, L., MATOUŠEK, J., RADOVÁ, V. *Mluvíme s počítačem česky*. První vydání. Praha: Academia, 2006. ISBN 80-200-1309-1.
- [10] SIGMUND, M., *Analýza řečových signálů*, Vysoké Učení Technické, Brno, 2000.
- [11] BARKER, J., *Computational Auditory Scene Analysis: Principles, Algorithms and Applications Chapter: 9*, University of Sheffield, Wiley and Sons, Prosinec 2005
- [12] HANSEN, J., *Analysis and Compensation of Speech under Stress & Noise for Enviromental Robustness in Speech Recognition*, p. 151-173, 1996. [online]. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S0167639396000507>
- [13] PSUTKA, J., HAJIČ, J., BYRNE, W. *The development of ASR for Slavic languages in the MALACH project*, IEEE, Montreal, 2004, s.749-752. [online]. Dostupné z: <https://ieeexplore.ieee.org/document/1326653>
- [14] RAJNOHA, J., POLLÁK, P. *ASR systems in Noisy Enviroment: Analysis and Solutions for Increasing Noise Robustness*. Czech Technical University, Prague, Czech Republic, 2011. [online]. Dostupné z: <https://dspace.vutbr.cz/handle/11012/56801>
- [15] BAIRD, C., SNYDER, F. *Terrain-aided altitude computations on the AFTI/F-16*. Harris Government Aerospace Systems Divison, Melbourne, Florida, 1990. [online]. Dostupné z: <https://ieeexplore.ieee.org/document/66217>
- [16] WEINSTEIN, J. C., *Opportunities for Advanced Speech Processing in Military Computer-Based Systems*, Lincoln Laboratory, 1991. [online]. Dostupné z : <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=118986>

- [17] WILLIAMSON, D. T., *Flight test results of itt VRS-1290 in NASA OV-10*, Alexandria, VA, p. 335-345, 1996. [online]. Dostupné z: [https://www.researchgate.net/publication/235062330\\_Flight\\_Test\\_Results\\_of\\_ITT\\_VRS-1290\\_In\\_NASA\\_OV-10](https://www.researchgate.net/publication/235062330_Flight_Test_Results_of_ITT_VRS-1290_In_NASA_OV-10)
- [18] ZHAO, Y. *Speech-recognition technology in health care and special-needs assistance*. IEEE Signal Processing Magazine, University of Missouri, June 2009. [online]. Dostupné z: <https://ieeexplore.ieee.org/abstract/document/4815547>
- [19] LECOUEUX, B., VACHER, M., PORTET, F., *Distant speech recognition for home automation: Preliminary experimental results in a smart home*, Speech Technology and Human-Computer Dialogue, 2011. [online]. Dostupné z: <https://ieeexplore.ieee.org/document/5940728>
- [20] RODGER, J. A., PENDHARKAR, P. C., *A field study of database communication issues peculiar to users of a voice activated medical tracking application*, Decision Support Systems 43, stránky 168–180, 2007. [online]. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S0167923606001229>
- [21] BARDRAM, J.E., BALDUS, H., FAVELA, J., *Pervasive computing in hospitals: chapter three*, p. 49-75, 2006. [online]. Dostupné z: <https://pdfs.semanticscholar.org/66e0/4b6c441a7cb6987251c042c82f2e9c4423ed.pdf>
- [22] VACHER, M., GUIRAND, N., SERIGNAT, J. F., FLEURY, A., NOURY, N., *Speech recognition in a smart home: Some experiments for telemonitoring*, Speech Technology and Human-Computer Dialogue, 2009. [online]. Dostupné z: <https://ieeexplore.ieee.org/document/5156185>
- [23] LU, G., HANKINSON, T., *A technique towards automatic audio classification and retrieval*, Monash University, Churchill, Australia, 1998. [online]. Dostupné z: <https://ieeexplore.ieee.org/document/770818>
- [24] HAN, Y., HYUN, J., JEONG, T., YOO, J. H., HONG, J. *A Smart Home Control System based on Context and Human Speech*. Division of IT Convergence Engineering, Pohang, South Korea, 2016. [online]. Dostupné z: <https://ieeexplore.ieee.org/document/7423313>
- [25] MCLOUGHLIN, I., SHARIFZADEH, H. R., *Speech Recognition for Smart Homes*, Singapore, 2008. [online]. Dostupné z: [https://www.researchgate.net/publication/221702135\\_Speech\\_Recognition\\_for\\_Smart\\_Homes](https://www.researchgate.net/publication/221702135_Speech_Recognition_for_Smart_Homes)
- [26] KLEINSCHMIDT, T., MASON, M., WONG, E., SRIDHARAN, S., *The Australian english speech corpus for in-car speech processing*, Queensland University of Technology, Brisbane, Australia, 2009. [online]. Dostupné z: <https://ieeexplore.ieee.org/document/4960549>
- [27] LEE, C., GAUVAIN, J., *Bayesian adaptive learning and MAP estimation of HMM*, Kluwer Academic Publishers, Boston, Massachusetts, USA, 1996.
- [28] SHADIEV, R., HWANG, W. Y., CHEN, N. S., HUANG, Y. M., *Review of Speech-to-Text Recognition Technology for Enhancing Learning*, Educational Technology and Society, November 2014.
- [29] WU, X. F., XU, J. C., *Using Speech Recognition Technology To Support Education For Deaf Students*, IEEE 978-1-4244-5265-1, Chengdu, China, 2010. [online]. Dostupné z: <https://ieeexplore.ieee.org/document/5477981>

- [30] STOUTEN, V., VAN HAMME, H., WAMBACQ, P., *Joint removal of additive and convolutional noise with model-based feature enhancement*, Katholieke Universiteit Leuven, Heverlee, Belgium, 2004. [online]. Dostupné z: <https://ieeexplore.ieee.org/document/1326144>
- [31] VASEGHI, S., V. *Advanced Digital Signal Processing and Noise Reduction. Fourth Edition*, Ltd. ISBN: 978-0-470-75406-1 [online]. 2008. Dostupné z: <http://sharif.ir/~bahram/sp4cl/MainReferences/advancedDigitalSignalProcessingAndNoiseReduction.pdf>
- [32] SOVKA, P., POLLÁK, P., *The study of speech/pause detectors for speech enhancement methods*, Conference paper, Czech Republic, Prague, 2014 [online]. Dostupné z: [https://www.researchgate.net/publication/221485906\\_The\\_study\\_of\\_speechpause\\_detectors\\_for\\_speech\\_enhancement\\_methods](https://www.researchgate.net/publication/221485906_The_study_of_speechpause_detectors_for_speech_enhancement_methods)
- [33] PSUTKA, J., *Komunikace s počítačem mluvenou řečí*. 1. vyd., Praha: Academia, 1995. ISBN 80-200-0203-0
- [34] ČERNOCKÝ, Jan. *Zpracování řečových signálů – studijní opora*. Vysoké učení technické v Brně [online]. 2006. Dostupné z: [http://www.fit.vutbr.cz/study/courses/ZRE/public/opora/zre\\_opora.pdf](http://www.fit.vutbr.cz/study/courses/ZRE/public/opora/zre_opora.pdf)
- [35] YOUNG, S., EVERMAN, G., a kol., *The HTK Book (for HTK Version 3.2)*, Cambridge University Engineering Department, 2002, s. 355. [online]. Dostupné z: <http://www.dsic.upv.es/docs/posgrado/20/RES/materialesDocentes/alejandrosViewgraphs/htkbook.pdf>
- [36] UPADHYAY, N., KARMAKAR, A., *An Improved Multi-Band Spectral Subtraction Algorithm for Enhancing Speech in Various Noise Environments*, Procedia Engineering, 2013 p. 312 – 321. [online]. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S1877705813016172>
- [37] CHAUDHARI, A., DHONDE, S., *A Review on Speech Enhancement Techniques*, International Conference on Pervasive Computing, 2015. [online]. Dostupné z: <https://ieeexplore.ieee.org/document/7087096/>
- [38] CHEN, J., a kol., *Speech Enhancement: Chapter one: Introduction*, ISBN: 9780128001394, Academic Press, Leden 2014.
- [39] VASEGHI, Saeed V., *Advanced Digital Signal Processing and Noise Reduction*. John Wiley and Sons, 2008.
- [40] SHARMA, D., KAUR, R., *Performance Analysis of Adaptive IIR Filter Using Sign Algorithm in LabVIEW*. In: 2015 2<sup>nd</sup> International Conference on Computing for Sustainable Global Development (INDIACom). Rajpura. IEEE, 2015. p. 1510-1513. [online]. Dostupné z: <https://ieeexplore.ieee.org/document/7100500>
- [41] MARTINEK, R. *Využití adaptivních algoritmů LMS a RLS v oblasti adaptivního potlačování šumu a rušení*. roč. 2013, č. 1. [online]. Dostupné z: [http://147.228.94.30/images/PDF/Rocnik2013/Cislo1\\_2013/r6c4c9.pdf](http://147.228.94.30/images/PDF/Rocnik2013/Cislo1_2013/r6c4c9.pdf)
- [42] MARTINEK, R., *Využití komplexních adaptivních metod zpracování signálů pro zpřesnění diagnostické kvality abdominálního fetálního elektrokardiogramu*, Ostrava, 2014, [online]. Dostupné z: <http://dspace.vsb.cz/handle/10084/105753>. Disertační práce.
- [43] MARTINEK, Radek, ŽÍDEK, Jan. Use of adaptive filtering for noise reduction in communications systems. 2010 International Conference on Applied Electronics (AE) 2010.

- Plzeň: Západočeská univerzita, 2010. ISBN 978-80-7043-865-7, ISSN 1803-7232. [online]. Dostupné z: <https://ieeexplore.ieee.org/abstract/document/5599607>
- [44] CICHOCKI, A., AMARI, S., *Adaptive Blind Signal and Image processing*, Japan, John Wiley & Sons, ISBN 0471 60791 6, 2002
- [45] JAGANNATH, D., SELVAKUMAR, A., *Issues and research on foetal electrocardiogram signal elicitation*, Biomedical Signal Processing and Control, 2014, p. 224-244. [online]. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S1746809413001560>
- [46] BACHARAKIS, E., NANDI, A., *Foetal ECG Extraction using Blind Source Separation Methods*, Signal Processing Division, Department of Electronic and Electrical Engineering, [online], Dostupné z: <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7083076&tag=1>
- [47] KONOPKA, O., *Analýza nezávislých komponent*, Czech Technical University in Prague. [online]. Dostupné z: <http://amber.feld.cvut.cz/bio/konopka/file/LBR-semestralka.pdf>
- [48] EKSLER, V., *Analýza hlavních komponent v problematice separace naslepo*, VUT Brno, 2005 [online] Dostupné z: <http://www.elektrorevue.cz/clanky/05029/index.html#kap7>
- [49] THOMPSEN, D. A. L., ANDERSEN, C. E., *Speech Enhancement and Noise-Robust Automatic Speech Recognition*, Signal Processing and Computing, Aalborg, 2015.
- [50] HYVARINEN, A., OJA, E., *Independent Component Analysis: A Tutorial*, 1999, [online]. Dostupné z: <http://www.cs.jhu.edu/~ayuille/courses/Stat161-261-Spring14/HyvO00-icatut.pdf>
- [51] UNGUREANU, M., BIGAN, C., STRUNGARU, R., LAZARESCU, V., *Independent Component Analysis Applied in Biomedical Signal Processing*, Bucharest, v.4, sec.2, 2004
- [52] MILITKÝ, J., MELOUN, M., *Metoda hlavních komponent a exploratorní analýza vícerozměrných dat*, [online]. Dostupné z: <https://meloun.upce.cz/docs/publication/127a.pdf>
- [53] LEACH, S. *Singular Value Decomposition - A Primer* [online]. Dostupné z: <http://people.csail.mit.edu/hasinoff/320/SingularValueDecomposition.pdf>
- [54] De MOOR, B., VANDEWALLE, J., *SVD-based methodologies for fetal electrocardiogram extraction*, Acoustics, Speech, Signal Processing, 1999, [online]. Dostupné z: [https://www.researchgate.net/publication/3858800\\_SVD-based\\_methodologies\\_for\\_fetal\\_electrocardiogram\\_extraction](https://www.researchgate.net/publication/3858800_SVD-based_methodologies_for_fetal_electrocardiogram_extraction)
- [55] Použitá zvuková karta Steinberg UR44. [https://www.steinberg.net/en/products/audio\\_interfaces/ur\\_series/models/ur44.html](https://www.steinberg.net/en/products/audio_interfaces/ur_series/models/ur44.html)
- [56] Použitý mikrofon Rode NT5. <http://www.ode.com/microphones/nt5>
- [57] ZEITNITZ, C. *WaveIO: A Soundcard Interface for LabVIEW*. Květen 2017. [online]. Dostupné z: <https://www.zeitnitz.eu/scms/waveio?PHPSESSID=ajv6kkd0fnuv6qaonak4mcrk90>
- [58] Popis Windows Speech Recognizeru. [https://en.wikipedia.org/wiki/Windows\\_Speech\\_Recognition](https://en.wikipedia.org/wiki/Windows_Speech_Recognition)
- [59] Odkaz pro stažení Windows Speech Recognizer. <https://www.microsoft.com/en-us/download/details.aspx?id=13045>
- [60] Odkaz pro stažení virtuálního kabelu VB-CABLE Virtual Audio Device. <https://www.vb-audio.com/Cable/>

## **Přílohy na CD**

- I. Elektronická verze diplomové práce.
- II. Zpracované programy:
  - Program pro nahrávání.
  - Program pro zpracovávání výsledků.
- III. Nahrávky použitých hluků a řeči:
  - Hluky z automobilu.
  - Hluky z letadla.
  - Hluky z domácnosti.
  - Hluky z továrny.
  - Vytvořené příkazy v bezdozvukové komoře.
- IV. Potřebné drivery k programům.